

Documentación Entrega 1 Proyecto – Diseño

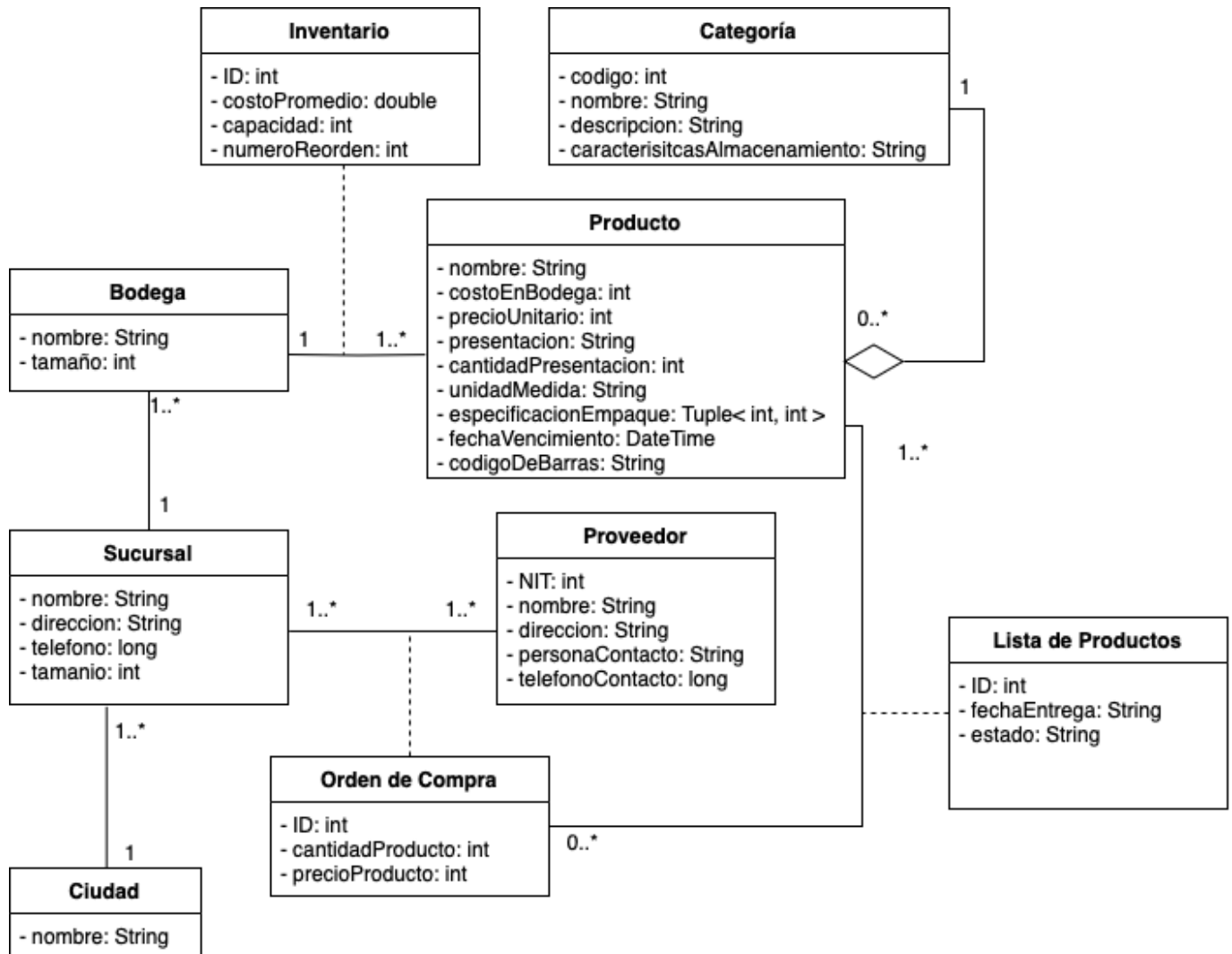
ISIS-2304 : Sistemas Transaccionales

Nikol T. De La Cruz D. – n.delacruz@uniandes.edu.co - 202317231

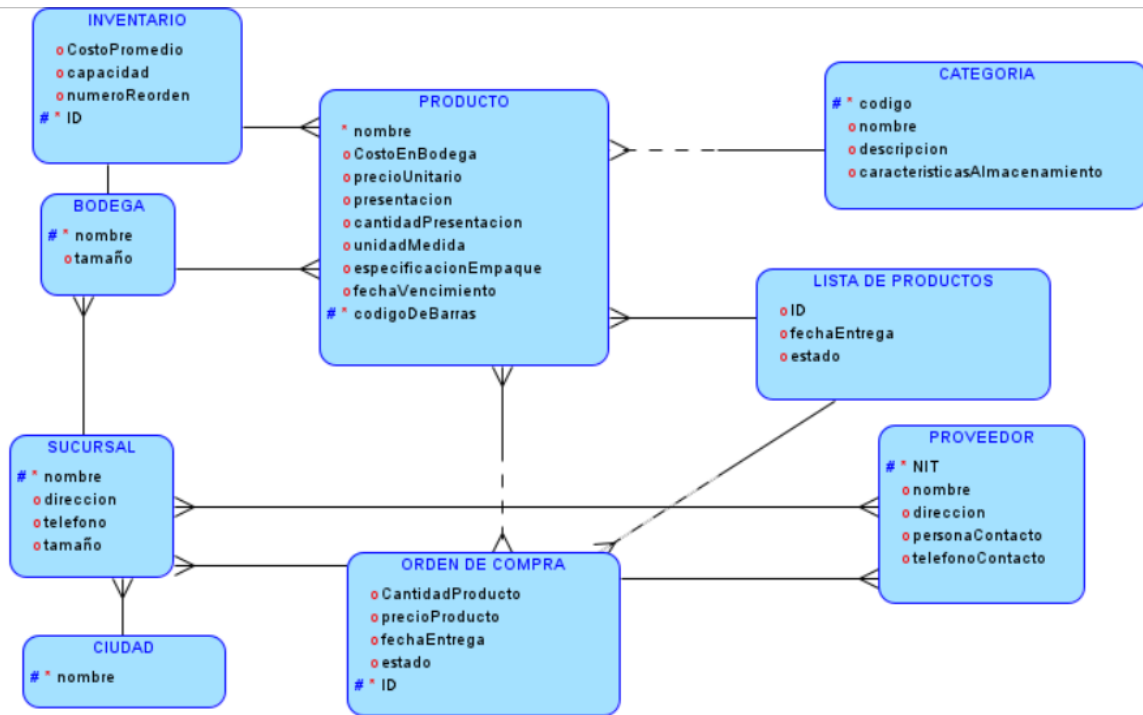
Juan D. Diaz V.– jd.diazv1@uniandes.edu.co - 202314374

Felipe A. Mesa N. – f.mesan@uniandes.edu.co – 202123007

1. Diagrama UML.



2. Diagrama Entidad/Relación.



3. Modelo de Datos Relacional.

Inventarios

ID	costoPromedio	capacidad	numeroReorden	bodegaAsignada	productoAsignado
PK, SA	DD, NN	NN, UA	NN, UA	NN, FK(bodega.nombre)	NN, FK(Producto.codigoDeBarras)

Ordenes de Compra

ID	cantidadProducto	precioProducto	sucursalAsignada	proveedorAsignado
PK	DD, NN	NN	NN, FK(Sucursal.nombre)	NN, FK(Proveedor.nit)

Categorías

codigo	nombre	descripción	característicasAlmacenamiento
PK	NN	NN, UA	NN

Productos									
CodigoDeBarras	costoEnBodega	precioUnitario	presentacion	cantidadPresentacion	unidadMedida	especificacionEmpaque	fechaVencimiento	nombre	categoriaAsignada
PK	NN	NN	NN,NC	NN	NN	NN	NN	NN,ND	FK(categoria.codigo), NN

ciudad
nombre
PK

Proveedores				
Nit	nombre	direccion	personaContacto	telefonoContacto
PK	NN	NN	NN	NN,ND

Sucursales				
nombre	direccion	telefono	tamaño	ciudadAsignada
PK	NN,ND	NN,ND	NN	NN,FK(Ciudad.nombre)

Bodegas		
nombre	tamaño	sucursalAsignada
PK	NN	FK(sucursal.nombre), NN

ListadeProductos				
codigoDeBarras	ordenDeCompraID	FechaEntrega	Estado	ID
FK(Producto.codigoDeBarras)	FK(orden de compra.ID)	NN	NN	PK

a. Justificación:

Para modelar el mundo de SuperAndes decidimos diseñar 9 tablas/relaciones diferentes. 6 (seis) para las entidades básicas y sus respectivos atributos, siendo estas

Ciudad, Sucursal, Proveedor, Producto, Categoría y Bodega; y 3 (tres) tablas/relaciones intermedias que representan la manera en la que estas 6 clases iniciales interactúan.

Para el caso de la interacción de Producto con Bodega, creamos una tabla intermedia llamada inventario que detalla elementos de la relación como el costo promedio de un producto dado en una bodega dada, su capacidad para dicho producto, entre otras. Por otro lado, para la relación entre sucursal y proveedor se creó una tabla/relación con el nombre Orden de Compra que detalla cosas como la cantidad de un producto y su precio establecido entre ambas partes. Finalmente se diseñó una tabla que representa la interacción entre una orden de compra y los productos que esta tiene, llamada lista de productos, y esta modela atributos como la fecha de entrega de dichos productos para una orden dada, así como su estado.

El uso del algoritmo de Chen se ve evidenciado principalmente en la decisión de diseño de implementar tablas intermedias que modelan las relaciones de muchos a muchos como fue descrito anteriormente, aunque también está presente en la escogencia de las restricciones en las columnas de las relaciones, como por ejemplo, la adición de la restricción NN (not null o no nulo) de las llaves foráneas (FK) de elementos que tienen obligatoriedad de relación uno a muchos, o de NN y ND (not duplicated o no duplicado) para aquellas relaciones que son obligatorias y estrictamente de cardinalidad 1 de ambos lados.

4. Normalización de las tablas.

5. Escenarios de prueba.

★ RF1:

- Pruebas de unicidad: Verificar que no se pueda insertar una tupla con clave primaria duplicada en la tabla.

Inserción: Ciudad (Nombre)

Valores: (Bogotá)

Se intenta insertar una tupla donde se agrega el nombre de una ciudad a la tabla que contiene los lugares en donde SuperAndes tiene sucursales.

Resultado: Si es una ciudad que no está repetida dentro de la tabla, la inserción es exitosa. De lo contrario será fallida porque dentro del atributo nombre no se pueden repetir valores (ND), el sistema debería generar un error indicando que ya existe ese valor.

★ **RF2:**

Prueba de Inserción de Datos: Verificar que se puede insertar sucursales y que están correctamente asociadas a una ciudad existente.

Inserción: Sucursales (nombre, dirección, telefono, tamaño, ciudadAsignada)

Valores: ('Sucursal 1', 'Calle 123', '1234567890', '300 m²', null)

aSe intenta crear una nueva sucursal con nombre 'Sucursal 1', dirección 'Calle 123', teléfono '1234567890', tamaño de '300 m²', pero sin asignar ninguna ciudad (ciudadAsignada es null).

Resultado : La inserción debería fallar porque el valor ciudadAsignada no puede ser nulo, ya que está restringido a ser NOT NULL y debe referenciar a una ciudad existente en la tabla Ciudad. El sistema debería generar un error indicando que se debe asignar una ciudad válida antes de poder crear la sucursal. Esto asegura que todas las sucursales estén correctamente asociadas a una ciudad existente y mantiene la integridad referencial.

★ RF3:

- **Pruebas de inserción de datos:** Verificar que se puede insertar sucursales y bodegas, y que las bodegas están correctamente asociadas a las sucursales.

Inserción: Bodegas(nombre, tamaño, sucursalAsignada)

Valores: (B1, 200 m, null)

Se quiere crear la bodega B1 con 200 m de tamaño, pero que no tiene ninguna sucursal.

La inserción debería fallar debido que este valor dentro de la tabla está restringido a no ser nulo, es decir, que antes de hacer una inserción de tuplas en la tabla Bodegas se debe crear una inserción en la tabla sucursal para evitar fallas.

★ RF4:

- Creación de proveedores:

Inserción: Proveedores(nit, nombre, dirección, personaContacto, telefonoContacto)

Valores: (22222222, Corona S.AS , Cra 17 #99-14, José Lopez, null)

Se quiere crear un nuevo proveedor con la información antes mencionada, sin embargo, aunque la mayoría de sus entradas son correctas nótese que uno de los valores es nulo.

El sistema debe arrojar un error debido que la información correspondiente a el telefono del contacto es nulo y eso no está permitido dentro de las tuplas que acepta la tabla de proveedores.

★ RF5:

- **CREAR Y LEER UNA CATEGORÍA DE PRODUCTO**

Prueba de Inserción de Datos: Verificar que se puede insertar una nueva categoría de producto y que la información se almacena correctamente.

- Inserción:** Categorías (codigo, nombre, descripcion, característicasAlmacenamiento)
- **Valores:** ('CAT001', 'Electrodomésticos', 'Productos electrónicos para el hogar', 'Almacenamiento en seco a temperatura ambiente')
 - **Explicación:** Se inserta una nueva categoría con código 'CAT001', nombre 'Electrodomésticos', descripción 'Productos electrónicos para el hogar' y características de almacenamiento 'Almacenamiento en seco a temperatura ambiente'. Esta inserción debe registrar toda la información necesaria para la categoría de producto según lo especificado en el enunciado.
 - **Resultado esperado:** La inserción es exitosa si todos los valores cumplen con las restricciones de integridad definidas, como que el código sea único (clave primaria) y que los campos nombre y descripción no sean nulos. El sistema debería almacenar correctamente esta información en la tabla Categorías.
 - **Prueba de Lectura de Datos:** Verificar que se puede recuperar la información de una categoría de producto existente.
 - **Consulta:** `SELECT * FROM Categorías WHERE codigo = 'CAT001' OR nombre = 'Electrodomésticos';`
Explicación: Se busca la categoría de producto con el código 'CAT001' o con el nombre 'Electrodomésticos'. La consulta debe devolver toda la información relacionada con esa categoría.
Resultado esperado: La consulta devuelve correctamente la información de la categoría 'Electrodomésticos', incluyendo el código, nombre, descripción y características de almacenamiento. Esto asegura que la categoría ha sido correctamente almacenada y puede ser recuperada según sea necesario

★ RF7:

- **CREAR UNA ORDEN DE COMPRA PARA UNA SUCURSAL**
Prueba de Inserción de Datos: Verificar que se puede crear una orden de compra y que los detalles están correctamente asociados a los productos y sucursales.
Inserción: OrdenesDeCompra (ID, cantidadProducto, precioProducto, sucursalAsignada, proveedorAsignado)

Valores: ('ORD001', 100, 50000, 'SUC001', 'PROV001')

Explicación: Se inserta una nueva orden de compra con ID 'ORD001', cantidad de producto 100, precio del producto 50000, asignado a la sucursal 'SUC001' y al proveedor 'PROV001'. Esto implica que la orden debe estar correctamente relacionada con los productos existentes y la sucursal correspondiente.

Resultado esperado: La inserción es exitosa si todos los valores cumplen con las restricciones de integridad, como que el ID sea único (clave primaria), y que las claves foráneas sucursalAsignada y proveedorAsignado existan en las tablas Sucursales y Proveedores, respectivamente (integridad referencial).

Prueba de Lectura de Datos: Verificar que se puede recuperar la información de una orden de compra existente.

Consulta: SELECT * FROM OrdenesDeCompra WHERE ID = 'ORD001';

Explicación: Se busca la orden de compra con ID 'ORD001'. La consulta debe devolver toda la información relacionada con esa orden, incluyendo la sucursal y el proveedor asignados.

Resultado esperado: La consulta devuelve correctamente la información de la orden de compra, incluyendo los detalles de la sucursal y el proveedor.