

Integrantes: Monica Diaz y Ana Charry

## Descripción del Proyecto - Sistema de Carrito de Compras

El proyecto tiene como objetivo desarrollar un sistema de carrito de compras utilizando una base de datos relacional que soporte dos roles principales: Administrador y Comprador. Los estudiantes deben implementar funcionalidades que permitan a los usuarios realizar operaciones como la gestión de productos, inventario y facturas, así como la compra de productos. El proyecto también requiere la creación de consultas SQL que involucren múltiples entidades mediante el uso de JOINS, además de la entrega de scripts completos en SQL para la creación (DDL) y manipulación (DML) de las tablas.

Este sistema se estructura para permitir la interacción eficiente entre los usuarios (administradores y compradores) y las entidades de la base de datos, lo que facilita el flujo de información para las compras en línea.

## Actores del Sistema

**Administrador** El Administrador tiene privilegios para gestionar los datos del sistema, incluyendo:

Crear, actualizar o eliminar productos disponibles en el sistema. Actualizar el inventario de productos.

Visualizar las facturas generadas por los compradores. **Comprador** El Comprador interactúa con el sistema de carrito de compras para:

Navegar por los productos disponibles. Agregar productos al carrito. Finalizar la compra generando una factura. **Base de Datos** La Base de Datos es un actor abstracto que representa la interacción entre las operaciones del sistema y el almacenamiento de datos. El sistema realiza consultas SQL a la base de datos para:

Recuperar información sobre productos. Actualizar inventarios. Generar facturas. Consultar datos entre múltiples tablas relacionadas.

## Casos de Uso

### Administrador

**Gestionar Productos:**

El administrador tiene la capacidad de crear, editar o eliminar productos dentro del sistema. Esta funcionalidad incluye definir el nombre del producto, su precio y una breve descripción. Se ejecutan comandos SQL como INSERT, UPDATE y DELETE sobre la tabla de productos. **Gestionar Inventario:**

El administrador puede actualizar las cantidades de los productos disponibles en el inventario. Esto implica ajustar el número de unidades disponibles y mantener el control sobre el stock de cada producto. Este proceso involucra operaciones UPDATE en la tabla de inventario. **Ver Facturas:**

El administrador puede consultar las facturas generadas por los compradores. Esto permite visualizar la información detallada de las compras, incluyendo los productos comprados, sus cantidades, el total pagado y la fecha de la transacción. Aquí se ejecutan consultas SELECT sobre las tablas de factura y detalle de factura, junto con posibles combinaciones de datos usando JOINS entre varias tablas.

# Comprador

## Navegar Productos:

El comprador puede ver el catálogo de productos disponibles para la compra, con detalles como el nombre, precio y descripción de cada producto. Esto se implementa con una consulta SELECT sobre la tabla de productos. Agregar al Carrito:

El comprador puede agregar productos a su carrito de compras, especificando las cantidades deseadas de cada artículo. El sistema actualiza los datos del carrito del comprador. En la base de datos, esto se refleja con una inserción o actualización en la tabla que gestione el carrito o los detalles de la factura. Generar Factura:

Al completar una compra, el comprador genera una factura que detalla todos los productos adquiridos, la cantidad de cada uno y el total de la compra. Esta funcionalidad involucra operaciones de INSERT en las tablas de Factura y Detalle Factura, para registrar la transacción y los productos comprados.

## codido wsd

```
@startuml
```

```
actor "Administrador" as Admin
actor "Comprador" as Buyer
actor "Base de Datos" as DB
```

```
rectangle SistemaCarritoDeCompras {
```

```
    usecase "Gestionar Productos" as UC1
    usecase "Gestionar Inventario" as UC2
    usecase "Ver Facturas" as UC3
```

```
    usecase "Navegar Productos" as UC4
    usecase "Agregar al Carrito" as UC5
    usecase "Generar Factura" as UC6
    usecase "Realizar Consultas SQL" as UC7
```

```
Admin -> UC1 : "Crear, Modificar, Eliminar"
Admin -> UC2 : "Actualizar Cantidades"
Admin -> UC3 : "Ver Detalles de Factura"
```

```
Buyer -> UC4 : "Ver Productos Disponibles"
Buyer -> UC5 : "Agregar Productos al Carrito"
Buyer -> UC6 : "Realizar Compra y Generar Factura"
```

```
UC1 --> DB : "Insert, Update, Delete en Productos"
UC2 --> DB : "Update en Inventario"
UC3 --> DB : "Select en Facturas y Detalles"
```

```
UC4 --> DB : "Select en Productos"
UC5 --> DB : "Update en Carrito"
UC6 --> DB : "Insert en Facturas y DetalleFactura"
```

```
UC7 --> DB : "JOINS entre Usuario, Producto, Inventario, Factura,
```

```
DetalleFactura"
}

@enduml
```

diagrama

