Proyecto de Diseño de Software (Avance 2)

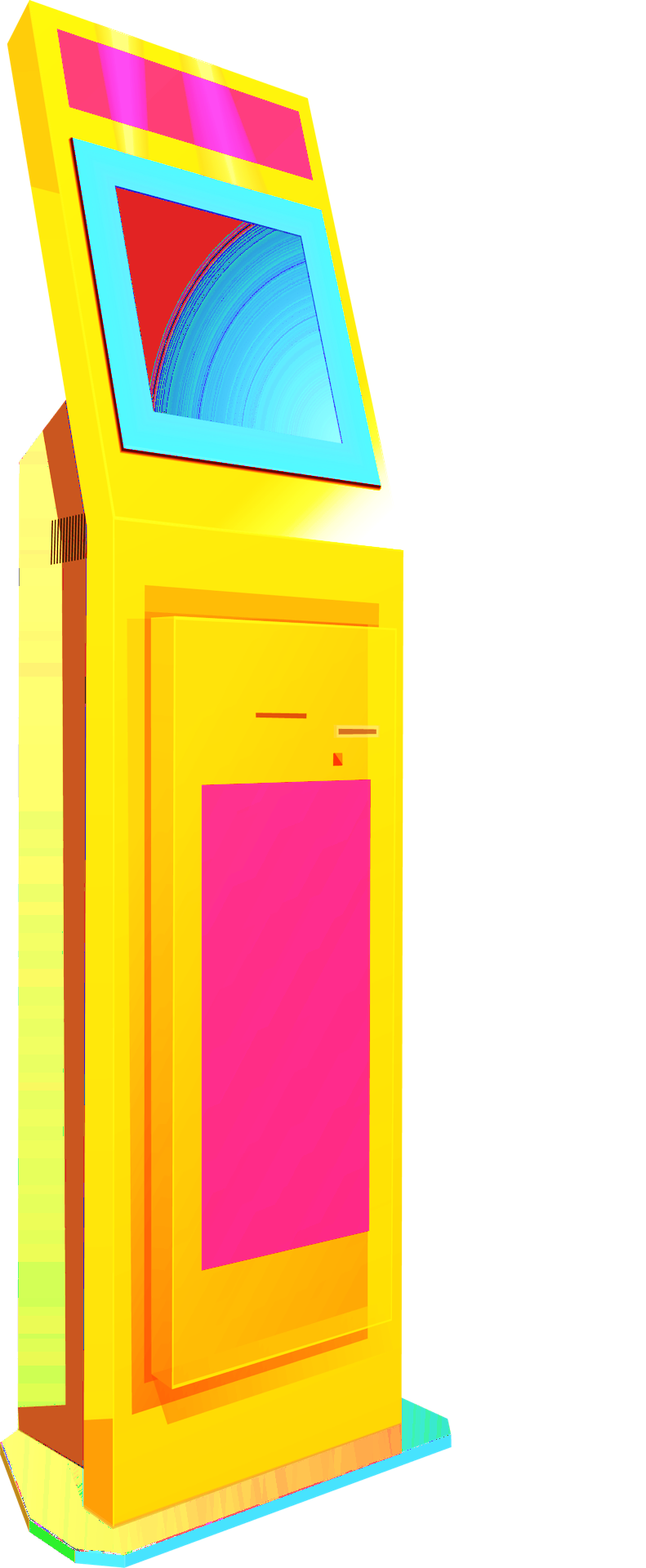
*Plataforma para el comercio de productos por parte de emprendedores politécnicos (PoliVentas)*

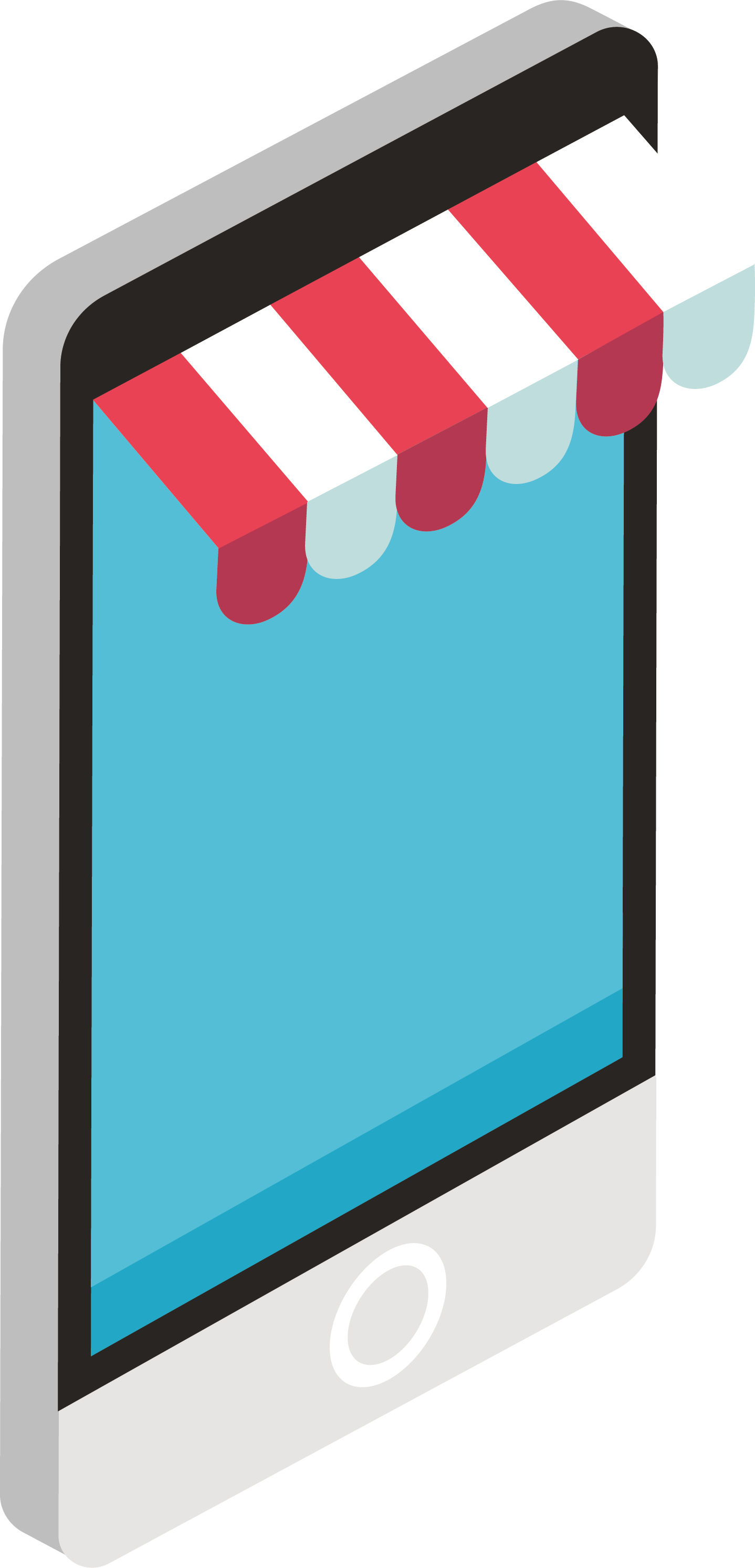
**Diseño de software**

##### Segundo Término 2018-2019

**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**

**Facultad de ingeniería en electricidad y computación**





 **Índice**

Poliventas

Integrantes 3

Titulo del Proyecto 3

Objetivo del Proyecto 3

Descripción del Sistema 3

Requerimientos 3

Diagrama de casos de usos 7

Justificación de los patrones de diseño 8

Diagrama ER de la base de datos 9

# Integrantes

* Luis A. Sarango-Parrales
* Harold Arturo Yepez Nieto
* Edison Raúl Barreiro Zambrano
* Anthony Gabriel Espinoza Suárez

# Título del Proyecto

Plataforma para el comercio de productos por parte de emprendedores politécnicos (PoliVentas)

# Objetivo del Proyecto

-Mejorar el proceso de comercialización de productos que los estudiantes politécnicos ofrecen.

# Descripción del Sistema

Una empresa que desea crear una plataforma que permita reunir en un solo lugar a varios estudiantes emprendedores que deseen vender sus productos y que puedan receptar el pago mediante dicha plataforma. Para el desarrollo de este sistema, se ha decidido contratar a persona con sólidos conocimintos en Diseño de Software para que diseñen una solución robusta y escalable. La plataforma propuesta debe utilizar una arquitectura MVC dentro de una aplicación desktop como plan piloto, después se decidirá la infraestructura adecuada para este tipo de aplicación.

El sistema utiliza una base de datos en donde se almacenan todos los datos de los usuarios, productos, ventas etc. Una persona podrá ingresar al sistema colocando su usuario y contraseña. Luego de verificar que estas credenciales pertenecen a un usuario válido del sistema (comprador, vendedor, administrador), se procederá a mostrar la interfaz de acuerdo con el rol. Si el usuario no pertenece al sistema, se podrá registrar llenando un formulario con su información personal.

El sistema tiene 3 roles; administrador, vendedor, comprador, con su respectivas capacidades se pueden realizar búsquedas de productos, ventas, compras; consultas de compras realizadas, etc. El administrador se encargará del ingreso de los 2 roles restantes y el vendedor también podrá ejercer el rol del comprador. Este vendedor, tiene una lista de productos propios.

El comprador, puede consultar productos y vendedores por diferentes filtros y campos, por el momento, se tiene disponible dos métodos de pago: efectivo o comprando saldo con un número asociado. Finalmente, se establecerá la fechaHora, lugar y el quiosco escogido entre el Comprador-Vendedor para completar la transacción. No obstante, en caso de no desear o cambiar el estado de la compraVenta se podrá anular. Además, el comprador será capaz de calificar al vendedor, junto a sus productos.

# Requerimientos

**1. Persistencia de datos**

El sistema debe utilizar una base de datos relacional (normalizada) en donde se almacenen todos los datos de los usuarios, productos, ventas etc. Para el informe se necesita que el diseño conceptual y diseño lógico (físico). El motor de base de datos puede ser el de su elección, pero debe estar instalado en otro servidor, distinto del que corre la aplicación java. Finalmente, se necesita datos de prueba, para poder realizar las búsquedas, los reportes, etc.

**2. Acceso al sistema**

Una persona podrá ingresar al sistema colocando su usuario y contraseña. Luego de verificar que estas credenciales pertenecen a un usuario válido del sistema (comprador, vendedor, administrador), se procederá a mostrar la interfaz de acuerdo con el rol. Si el usuario no pertenece al sistema, se podrá registrar llenando un formulario con su información personal incluyendo nombres, apellidos, teléfono, si tiene Whatsapp o no, email, dirección, cédula, matrícula, perfil(vendedor/comprador). Los administradores pueden cambiar el rol de una persona a cualquiera de los disponibles (incluido administrador). En el futuro podrían existir más roles. Todos los usuarios del sistema son estudiantes.

**3. Opciones de usuario**

Luego de que el usuario ha ingresado con unas credenciales válidas, se deberá mostrar el menú correspondiente a su rol. Para la GUI considere que esta aplicación se utilizará con una PC con pantalla táctil, puesto que la idea es que funcione como un quiosco de compras donde acepta dinero físico.

Si es **Comprador**, se le mostrarán las siguientes opciones:

* 1. 1) Búsqueda

a. Sencilla

* 1. b. Avanzada
  2. 2) Mis pedidos

a. Compras pendientes

b. Historial

* 1. 3) Artículos más buscados

4) Nuevos artículos

5) Cerrar sesión (Regresar al login)

Si es **Vendedor**, se le mostrarán las opciones de **comprador** y **también** las que están a continuación:

* 1. 1) Mis ventas
     1. a. Ventas pendientes
     2. b. Resumen de ventas
  2. 2) Mis productos
  3. 3) Cerrar sesión (Regresar al login)

En caso de que sea un **administrador**, se le mostrarán las siguientes opciones:

1) Administrar usuarios

* 1. 2) Búsqueda

1. a. Sencilla
2. b. Avanzada
   1. 3) Compras
3. a. Pendientes de entrega
4. b. Anuladas
5. c. Exitosas
   1. 4) Administrar Productos
   2. 5) Cerrar sesión (Regresar al login)

**4. Búsqueda**

El sistema debe buscar el texto ingresado por el usuario pero que no diferencie mayúsculas de minúsculas ni acentos o signos de puntuación. Dicha búsqueda se realiza tanto en el nombre como en la descripción del producto, pero como mínimo se debe ingresar 3 caracteres distintos de espacio y se verificará que esté contenido en los campos antes mencionados.

La búsqueda avanzada permite realizar filtros por costo, vendedores, categorías de los productos, calificación del producto, calificación del vendedor. Tome en cuenta que se pueden aplicar varios filtros simultáneamente.

Cada artículo mostrado debe tener nombre, categoría, precio, tiempo máximo de entrega y calificación del producto y del vendedor (Escala de 5 estrellas para cada uno). Aquí mismo deberá permitir comprar dicho artículo. El pago se debe realizar mediante compra de saldo utilizando una plataforma virtual (App de celular que ya está disponible) y que está vinculado con el número de celular del usuario actual o también puede pagar con efectivo en los quioscos donde corre la plataforma. El usuario debe elegir una sola forma de pago. Una vez realizado el

pago, se debe notificar automáticamente al vendedor de dicho producto, para que este realice la entrega.

**5. Mis pedidos**

Esta opción muestra el listado de los pedidos pendientes, es decir, aquellos que aún no han sido recibidos. Además, se podrá visualizar un historial de todos los pedidos realizados, organizarlos por fecha, por costo y filtrarlos por nombre del producto o por vendedor. Finalmente, esto permitirá generar un PDF con la información filtrada, la cual será enviada al correo electrónico del usuario actual.

Aquí se tiene la posibilidad de calificar el producto y el vendedor utilizando estrellas. Además, se puede indicar si el producto fue entregado a tiempo o no, incluso se puede anular el pedido. Finalmente, cuando el vendedor llega con el producto pedido, el comprador debe indicar que recibió el producto, caso contrario el vendedor no le entregará dicho producto y venta se anula.

**6. Mis ventas**

Esta opción muestra el listado de las ventas, se puede elegir entre las pendientes, anuladas y exitosas. Aquí el vendedor puede anular una venta o incluso ver en un mapa, donde debe realizar la entrega de dicho artículo. Las entregas se las realiza junto a los quioscos y ambos (comprador y vendedor) deben esperar por lo menos 15 minutos de la hora pactada para dar tiempo a que llegue el otro. Después del tiempo de espera se podrá anular la venta.

**7. Mis productos**

El vendedor podrá realizar las operaciones CRUD sobre sus productos, sin embargo, la eliminación es de tipo lógica y no podrá ser deshecha por él, sino por el administrador.

**8. Artículos más buscados**

Este listado debe mostrar el nombre y el precio de los 15 artículos más buscados en forma de tabla, al dar clic sobre uno de los artículos, el sistema debe verificar automáticamente si el usuario ha ingresado al sistema o es un usuario desconocido. En caso de que haya ingresado al sistema como comprador o vendedor, le deberá aparecer la ventana de consulta del producto y el botón para comprar, caso contrario, deberá mostrarse la interfaz de Login.

**9. Salir del sistema**

Al salir del sistema debe regresar a la pantalla de inicio. En la pantalla de inicio se muestran dos listados, Los artículos más buscados y los nuevos artículos ingresados a la plataforma. Estos listados solo permiten ver la información de los artículos, pero no se pueden comprar mientras el usuario no ha ingresado y verificado su cuenta.

**10. Administrar usuarios y productos**

Los administradores podrán realizar las operaciones CRUD sobre usuarios y productos, sin embargo, la eliminación debe ser de tipo lógico, es decir; ningún registro se elimina realmente de la base, sino que se deshabilita para que puedan ser consultados o buscados por los otros usuarios.

**11. Pagos**

La plataforma permite realizar pagos en efectivo o con saldo adquirido de forma electrónica, para los usuarios esto es transparente, simplemente compran con el saldo disponible. Sin embargo, se está analizando otros medios de pago electrónico como tarjeta de crédito Visa.

**12. Reportes**

a. La cantidad de ventas semanales del año en curso, por tipo de pago y vendedor.

b. Los 10 artículos más vendidos en los últimos 3 meses.

La empresa desea analizar ciertos reportes que necesitará en el futuro para realizar ofertas y tomar decisiones sobre las nuevas funcionalidades que pueda tener la plataforma, por lo tanto, ha pedido que, junto con la documentación del sistema, muestre dos consultas SQL:

**Diagrama de casos de usos**

****

**Patrón utilizado – Justificación**

**Fabrica Abstracta:**

Este patrón, permite escribir aplicaciones que son más flexibles respecto de los tipos a utilizar, difiriendo la creación de las instancias en el sistema a subclases que pueden ser extendidas a medida que evoluciona el sistema. Permite también encapsular el conocimiento referente a la creación de objetos, haciendo también que el diseño sea más adaptable a cambio de sólo un poco más de complejidad.

Se detectaron dos familias. Persona con sus productos Administrador, Comprador y Vendedor. Por otro lado, la familia Operación con sus productos Compra y Entrega.

**Strategy:**

Este patrón, permite mantener un conjunto de algoritmos de entre los cuales el objeto cliente puede elegir aquel que le conviene e intercambiarlos dinámicamente según sus necesidades.

Al existir una compra, consecuentemente deberá existir un pago, por esta razón hay dos formas(dos estrategias) de realizar la misma acción que es pagar, mediante un pago en efectivo o pago electrónico (comprando saldo) y muy posiblemente se agregen más en el futuro.

**Singleton:**

Es evidente que debido a los requerimientos de la app se necesita una conexión al motor de base de datos escogido. Por tanto, para llevar a cabo esta tarea, se puede crear una solo instancia para el manejo de conectar y desconectar la app al motor de BD, en vez de instanciar a cada momento un objeto de la conexión para cada operación en el manejo de datos.

**DAO:**

Para el manejo de datos (operaciones CRUD) por parte de EstudianteDAOImpl, ProductoDAOImp, CompraDAOImpl. Se crearán estos objetos DAO que encapsulan la tecnología empleada para acceder a la capa de datos, separando completamente la lógica de negocio de la lógica de acceso a datos. De esta forma se garantiza la portabilidad del sistema en los diferentes motores de base de datos, evitando mezclar capas.

**Diagrama ER de la base de datos**

