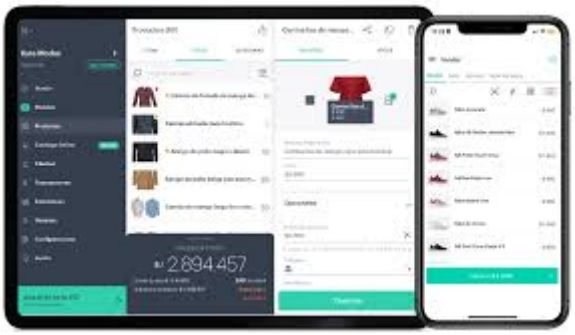
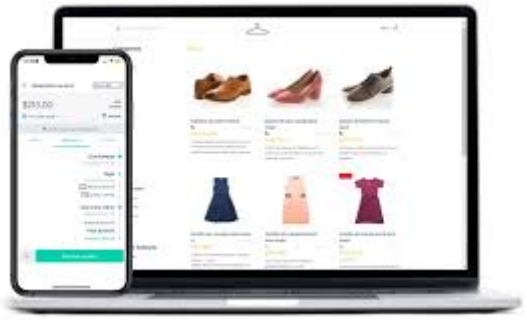
Daniel Carrasco Espinosa 28/05/2025

Aplicación: Tienda de ropa.





ÍNDICE

Clase Cliente

Descripción: Clase para representar un cliente que entra a realizar una compra en la tienda

Atributos:

idCliente: Identificador único del cliente.
nombre: Nombre completo del cliente.
correo: Correo electrónico del cliente.
dirección: Dirección física del cliente.
teléfono: Teléfono del cliente.

Métodos:

realizarOrden(orden: Orden): void: Permite al cliente realizar una orden.
actualizarDatos(nombre: String, correo: String, direccion: String, telefono: String): void: Actualiza los datos del cliente.

Clase Orden

Descripción: Clase para representar una orden de compra de algún cliente.

Atributos:

idOrden: Identificador único de la orden.
fecha: Fecha de creación de la orden.
estado: Estado de la orden(aprobada ,denegada o pendiente).
productos: Lista de productos asociada a la orden.
factura: Factura total de la orden.

Métodos:

calcularFacturaTotal(): double: Calcula la factura total basado en los productos.
agregarProducto(producto: Producto): void: Añade un producto a la orden.
eliminarProducto(producto: Producto): void: Elimina un producto de la orden.
cambiarEstado(nuevoEstado: String): void: Modifica el estado de la orden.
getFactura() : void: Nos devuelve una factura.

Clase Producto

Descripción: Clase para representar productos que están en la tienda.

Atributos:

idProducto: Identificador único del producto.
nombre:Nombre del producto.
precio: Precio(€) del producto en específico.
categoría: Categoría a la que pertenece el producto.

Métodos:

mostrarDetalles(): void: Muestra los detalles del producto.
actualizarPrecio(nuevoPrecio: double): void: Actualiza el precio del producto por si existe alguna oferta o simplemente el precio del producto sube o baja.
getPrecio(): void : nos devuelve un precio de un producto.

Clase Categoría

Descripción: Clase para representar las diversas categorías en las cuales se clasifican los productos.

Atributos:

idCategoria: Identificador único de la categoría.
nombre:Nombre de la categoría del producto.
productos: Lista de productos.

Métodos:

asignarProducto(producto: Producto): void: Asocia un producto con la categoría.
mostrarProductos(): void: Muestra todos los productos de esta categoría.

Clase Ropa

Descripción: Clase para representar la diferente ropa que vende la tienda.

Atributos:

talla: Talla del producto.
color: Color del producto.
tela: tipo de tela de la cual está compuesta la ropa.

Métodos:

mostrarDetalles(): void: Sobreescribe el método para incluir detalles específicos de la ropa.

Clase Pago

Descripción: Clase para representar el pago de la factura de los productos.

Atributos:

idPago: Identificador único del pago.
factura: Factura del pago.
metodoPago :Método por el cual se va a pagar (Efectivo o tarjeta).
fechaPago: Fecha en la cual se realizó el pago.

Métodos:

mostrarDetalles(): void: Muestra los detalles del producto.
procesarPago(): boolean: Muestra los detalles del pago (verdadero si está pagado o falso si todavía no se ha hecho la transacción.
isPagado(): boolean: devuelve true o falso para ver si se ha efectuado el pago correctamente.

Clase APP

Descripción: Clase Main para probar los métodos ,crear objetos…

Clase AppTest

Descripción: Es una clase de prueba unitaria utilizando JUnit, una de las bibliotecas más comunes para pruebas en Java.

Clase ClienteTest

Descripción: Es una clase de prueba unitaria escrita con JUnit que verifica el correcto funcionamiento del método actualizarDatos de la clase Cliente.

Clase OrdenTest

Descripción: Es una prueba unitaria con JUnit que verifica que la suma del total de una orden (factura) se actualiza correctamente cuando se agregan productos.

Clase PagoTest

Descripción: Es una prueba unitaria con JUnit que verifica si el método procesarPago() de la clase Pago.

Clase ProductoTest

Descripción: Es una prueba unitaria con JUnit que verifica que el método getPrecio() de la clase Producto devuelve correctamente el precio del producto.

El diagrama de clases se encuentra en otro archivo .dia.