



PROYECTO FINAL

Sistema de Facturacion SuperMaxi en Loja

Aaron Robles - Fernando Ocampo

INTRODUCCION

En el actual contexto de la industria minorista, la gestión eficiente de las operaciones de facturación es fundamental para garantizar la satisfacción del cliente y el éxito comercial. El SuperMaxi en Loja, como parte de una cadena de supermercados reconocida a nivel nacional, enfrenta desafíos específicos en su proceso de facturación debido a la complejidad de su operación y la necesidad de mantener altos estándares de servicio.

El objetivo de este proyecto es desarrollar un Sistema de Facturación personalizado y eficiente para el SuperMaxi en Loja, que aborde las problemáticas actuales relacionadas con la gestión de productos, el cálculo preciso de facturas y la generación de estadísticas de ventas. Este sistema se diseñará para optimizar los procesos internos del supermercado, mejorando la precisión, la velocidad y la eficiencia en el manejo de transacciones de venta.

OBJETIVOS DEL PROYECTO

- **Desarrollar un Sistema de Facturación:** Crear un sistema completo que abarque desde la gestión de productos hasta la generación de facturas y estadísticas de ventas, con el fin de automatizar y optimizar el proceso de facturación en el SuperMaxi en Loja.
- **Satisfacción del Cliente:** Priorizar la experiencia del cliente al garantizar un proceso de facturación rápido, preciso y sin problemas, lo que contribuirá a la fidelización de los clientes y al fortalecimiento de la reputación del

PROBLEMÁTICA DEL PROYECTO

El SuperMaxi en Loja enfrenta diversos desafíos en su proceso de facturación que afectan su eficiencia operativa y la experiencia del cliente. Algunas de las principales problemáticas identificadas son las siguientes:

- 1. Procesos Manuales y Propensos a Errores:** El proceso actual de facturación depende en gran medida de métodos manuales y sistemas poco integrados, lo que aumenta la probabilidad de errores humanos y dificulta la consistencia en la gestión de transacciones de venta.
- 2. Error en las Facuras:** La falta de un mecanismo automatizado para calcular las facturas, considerando descuentos, promociones y otros factores variables, conlleva a errores en los montos totales facturados y genera inconvenientes para los clientes y el personal de caja.
- 3. Dificultades en la Generación de Estadísticas:** La ausencia de herramientas adecuadas para analizar y generar estadísticas detalladas sobre las ventas, los productos más vendidos y las tendencias del mercado, limita la capacidad de los gerentes para tomar decisiones estratégicas informadas.

FRAGMENTOS IMPORTANTES

```
package Model;

import java.io.Serializable;
import java.text.SimpleDateFormat;
import java.util.Date;
public class Producto implements Serializable{

    private String nombre;
    private String categoria;
    private double precioNormal;
    private double precioPromocional;
    private int cantidadEnStock;
    private Date fechaCaducidad;

    private static final SimpleDateFormat fecha = new SimpleDateFormat(pattern: "dd/MM/yyyy");
```

Esta Parte del código implementa la interfaz serializable y además define una variable fecha que es una instancia de la clase SimpleDateFormat la cual se utiliza para ingresar una fecha en este caso de caducidad del producto ingresado por el usuario

FRAGMENTOS IMPORTANTES

```
public void agregarProducto(Producto producto) {  
    this.productos.add(e: producto);  
    if (producto.getCantidadEnStock() > 0 && producto.getFechaCaducidad().after(new Date())) {  
        this.total += producto.getPrecioNormal();  
    } else {  
        this.total += producto.getPrecioPromocional();  
    }  
}
```

En este Fragmento de código es un método llamado agregarProducto, en la condición verifica si el producto tiene una cantidad en stock mayor que cero y si su fecha de caducidad es posterior a la fecha actual, esto nos ayuda a verificar si el producto está a la venta y aún no se ha caducado.

FRAGMENTOS IMPORTANTES

```
System.out.println(x: "Ingrese la fecha de caducidad del producto (dd/MM/yyyy):");
String fechaCaducidadStr = scanner.nextLine();
Date fechaCaducidad = null;
try {
    fechaCaducidad = fecha.parse(source: fechaCaducidadStr);
} catch (ParseException e) {
}

return new Producto(nombre, categoria, precioNormal, precioPromocional, cantidadEnStock, fechaCaducidad);
}
```

En este Fragmento de código permite al usuario ingresar la fecha de caducidad del producto y la convierte en un formato fecha, dentro del TryCatch se hace la conversión la cadena de texto ingresada por el usuario en un objeto llamado Date, la llamada al método "parse" de la instancia SimpleDateFormat, si la cadena ingresada por el usuario no está en el formato especificado, se captura una excepción de tipo "ParseException".

FRAGMENTOS IMPORTANTES

```
public void mostrarInventario() {
    for (Producto producto : inventario) {
        consola.mostrarMensaje(mensaje:producto.toString());
    }
}

private void cargarInventario() {
    try (ObjectInputStream Invent_ario = new ObjectInputStream(new FileInputStream(name: INVENTARIO_FILE))) {
        inventario = (List<Producto>) Invent_ario.readObject();
    } catch (FileNotFoundException e) {
        consola.mostrarMensaje(mensaje:"No se encontro, el archivo de inventario. Se creara; uno nuevo.");
    } catch (IOException | ClassNotFoundException e) {
    }
}

private void guardarInventario() {
    try (ObjectOutputStream G_inventario = new ObjectOutputStream(new FileOutputStream(name: INVENTARIO_FILE))) {
        G_inventario.writeObject(obj:inventario);
    } catch (IOException e) {
    }
}

private void guardarFactura() {
    try (ObjectOutputStream G_Factura = new ObjectOutputStream(new FileOutputStream(name: FACTURA_FILE))) {
        G_Factura.writeObject(obj:factura);
    } catch (IOException e) {
    }
}
```


FRAGMENTOS IMPORTANTES

En mostrar Inventario recorre toda la lista de Productos y muestra cada producto en la consola utilizando el método toString

En Cargar Inventario este método carga el inventario utilizando el ObjectInputStream, primero se intenta abrir el archivo desde el inventario utilizando ObjectInputStream y leer el objeto de la lista de Productos, si el archivo no se encuentra mostrara un mensaje de "No se encontró el Archivo"

En guardar el archivo utilizan ObjectOutputStream, para escribir la factura en el archivo de factura.

RESULTADOS

Durante el desarrollo del proyecto de Sistema de Facturación para SuperMaxi en Loja, se lograron alcanzar varios resultados significativos que abordaron las problemáticas identificadas y mejoraron la eficiencia y la experiencia general del usuario. A continuación se detallan algunos de los resultados más relevantes:

1. Implementación del Sistema de Facturación: Se desarrolló e implementó con éxito un sistema de facturación integral que automatiza los procesos de gestión de productos, cálculo de facturas y generación de estadísticas de ventas.
2. Mejora en la Eficiencia Operativa: El nuevo sistema permitió agilizar significativamente el proceso de facturación, reduciendo el tiempo requerido para completar transacciones de venta y minimizando la incidencia de errores.

```
2,50
Ingrese el precio promocional del producto:
3,00
Ingrese la cantidad en stock del producto:
5
Ingrese la fecha de caducidad del producto (dd/MM/yyyy):
01/02/2030
Producto agregado exitosamente!
Producto{nombre='Jabon', categoria='Limpieza', precioNormal=2.43, precioPromocional=2.5, cantidadEnStock=4, fechaCaducidad=21/02/2004}
Producto{nombre='shampoo', categoria='vivienda', precioNormal=2.5, precioPromocional=3.0, cantidadEnStock=2, fechaCaducidad=21/03/2001}
Producto{nombre='Escoba', categoria='Vivienda', precioNormal=2.5, precioPromocional=3.0, cantidadEnStock=5, fechaCaducidad=01/02/2030}
Total de la factura: 8.0
BUILD SUCCESSFUL (total time: 29 seconds)
```

CONCLUSIONES

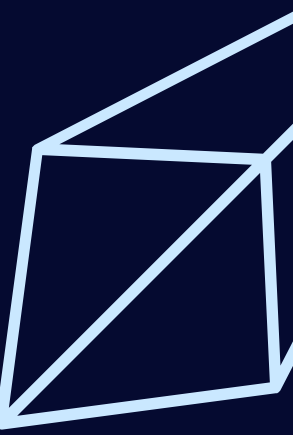
Durante el desarrollo del proyecto de Sistema de Facturación para SuperMaxi en Loja, se enfrentaron diversas dificultades que requirieron soluciones creativas y un enfoque colaborativo. A continuación se detallan algunas de las conclusiones más relevantes derivadas de esta experiencia:

1. **Dificultades al realizar código:** Durante el proceso de desarrollo, nos encontramos con desafíos técnicos relacionados con la integración de diferentes componentes del sistema y la gestión de grandes volúmenes de datos. Sin embargo, mediante una planificación cuidadosa y una resolución proactiva de problemas, pudimos superar estas dificultades y avanzar con éxito en el proyecto.

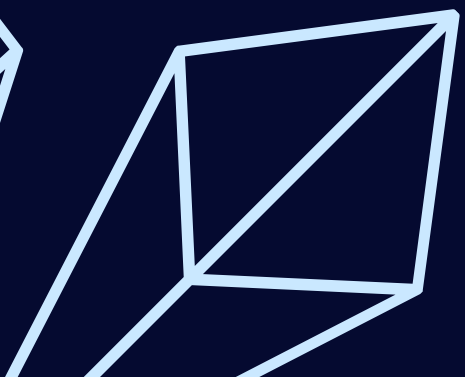
CONCLUSIONES

- **Experiencias Relevantes:** Durante el desarrollo del proyecto, tuvimos la oportunidad de trabajar en un entorno simulado que replicaba las condiciones del mundo real en un supermercado, lo que nos brindó una experiencia valiosa en la aplicación de soluciones prácticas a problemas complejos. Además, la interacción con el personal del SuperMaxi en Loja nos permitió obtener información directa sobre sus necesidades y desafíos específicos, lo que enriqueció nuestro enfoque de desarrollo.
- **Sugerencias para Futuros Proyectos:** Basándonos en nuestra experiencia, recomendamos a futuros equipos de desarrollo la importancia de realizar una investigación exhaustiva de los requisitos del cliente y mantener una comunicación abierta y continua con todas las partes interesadas.

DIFICULTADES:



- **Complejidad del Dominio del Problema:** La naturaleza compleja del proceso de facturación en un supermercado, que involucra múltiples variables y regulaciones, representó un desafío inicial para comprender completamente los requisitos del sistema y definir su alcance de manera efectiva.
- **Integración de todas las Clases:** La integración de todas las clases fue muy complicado ya que unir todo el código, es muy abstracto de realizar pero con trabajo y esfuerzo pudimos realizar el código planeado.
- **Manejo de Datos:** El manejo eficiente de grandes volúmenes de datos de productos y transacciones de venta planteó desafíos en términos de rendimiento y escalabilidad del sistema, especialmente durante la generación de estadísticas y análisis de datos.



The background is a solid dark blue color. On the left and right sides, there is a decorative pattern of bright blue lines forming a hexagonal grid. The lines are of medium thickness and intersect to create a series of hexagons of varying sizes. The pattern is more dense on the left and right edges and fades towards the center where the text is located.

GRACIAS