

**UNIVERSIDAD CATÓLICA BOLIVIANA “SAN PABLO”  
LA PAZ  
CARRERA DE SISTEMAS**



**Sistema de Administración de una Cafetería  
Programación II**

**Participante**

Gonzalo Gabriel Apaza Ticona

**2025**

# INDICE

RESUMEN .....	1
1. INTRODUCCIÓN .....	2
1.1 Contexto del Proyecto.....	2
2. OBJETIVOS .....	2
2.1 Objetivo General .....	2
2.2 Objetivos Específicos.....	2
3. RECOLECCIÓN DE INFORMACIÓN .....	3
4. ALCANCE DEL SISTEMA.....	3
5. ANÁLISIS DE REQUERIMIENTOS .....	4
5.1 Identificación de Actores.....	4
5.2 Casos de Uso Principales .....	6
5.3 Lista de Requisitos Funcionales .....	8
6. PLANIFICACIÓN Y GESTIÓN DEL PROYECTO .....	11
6.1 Cronograma de Desarrollo.....	11
6.2 Asignación de Roles .....	12
7. MODELADO DEL SISTEMA .....	12
7.1 Diagrama de Clases (UML).....	12
7.2 Diagrama Entidad - Relación .....	13
8. DESCRIPCIÓN DE LA ARQUITECTURA .....	14
8.1 Arquitectura en Tres Capas.....	14
8.2 Capa de Presentación (PRESENTACIÓN) .....	14
8.3 Capa de Lógica de Negocio (LOGICA).....	15
8.4 Capa de Persistencia (DATOS) .....	16
9. DESARROLLO TECNOLÓGICO.....	17
9.1 Tecnologías Utilizadas .....	17
10. MANUAL DE USUARIO.....	18
10.1 Introducción .....	18
10.2 Inicio del Sistema.....	18
10.3 Panel Principal (Selección de Módulos) .....	18
10.4 Módulo de Administración.....	19
Gestión de Productos.....	19
Gestión de Categorías .....	19

Promociones y Ofertas.....	20
Reportes (Administración) .....	20
Dashboard .....	20
Gestión de Usuarios .....	20
Respaldo de Base de Datos .....	21
10.5 Punto de Venta.....	21
Nueva Venta.....	21
Sistema de Cobranza .....	21
Gestión de Clientes .....	21
Reportes .....	22
10.6 Módulo de Reservas de Mesa .....	22
10.7 Búsquedas y Filtros.....	22
10.8 Cerrar Sesión o Salir .....	22
11. CONCLUSIONES Y APRENDIZAJES.....	23
11.1 Conclusiones.....	23
11.2 Aprendizajes.....	23
12. ANEXOS .....	24
12.1 Capturas GUI .....	24
12.2 Capturas del código.....	30
13. LINK DEL REPOSITORIO.....	32

## **RESUMEN**

El presente informe documenta el análisis, diseño e implementación de un Sistema de Gestión de una Cafetería desarrollado utilizando Java para la interfaz gráfica y MySQL como sistema de gestión de base de datos. El sistema integra módulos para la administración de usuarios, menú, reservas, pedidos, cobranza, inventario, promociones y reportes estadísticos.

El proyecto se desarrolló bajo una arquitectura de tres capas (Presentación, Lógica y Datos), lo que permite garantizar una adecuada separación de responsabilidades, escalabilidad y facilidad de mantenimiento. Durante el proceso de desarrollo se aplicaron prácticas de ingeniería de software como levantamiento de requerimientos, modelado UML, diseño relacional, pruebas unitarias e integración modular.

El sistema final ofrece una solución eficiente para automatizar las operaciones principales de una cafetería, mejorando la gestión interna, reduciendo errores humanos y facilitando la toma de decisiones basada en datos.

## **1. INTRODUCCIÓN**

En el entorno actual, las cafeterías requieren herramientas tecnológicas capaces de optimizar sus procesos internos y mejorar la experiencia del cliente. La gestión manual de reservas, pedidos, inventario y cobranza suele generar demoras, errores y dificultades en el control operativo, especialmente en negocios con alta demanda.

El presente proyecto desarrolla un sistema informático para la administración integral de una cafetería, implementado bajo una arquitectura modular que facilita su mantenimiento, escalabilidad y personalización. A través de este sistema, se busca automatizar las tareas más relevantes del negocio, centralizar la información y permitir una operación más eficiente para el personal del establecimiento.

### **1.1 Contexto del Proyecto**

El proyecto surge ante la necesidad de contar con una herramienta que permita gestionar de manera más eficiente los procesos diarios de una cafetería, reemplazando métodos manuales o sistemas parciales que no cubren completamente las operaciones del negocio. La solución propuesta está orientada a establecimientos que requieren un control preciso del menú, inventario, pedidos, reservas y cobranza, con especial énfasis en la facilidad de uso y la confiabilidad del sistema.

## **2. OBJETIVOS**

### **2.1 Objetivo General**

Desarrollar un sistema informático integral que automatice y optimice los procesos operativos de una cafetería, mejorando la eficiencia del personal, la experiencia del cliente y facilitando la toma de decisiones gerenciales.

### **2.2 Objetivos Específicos**

- **Implementar un sistema de autenticación** sólido con control de acceso basado en roles.
- **Desarrollar módulos funcionales** para gestión de menú, pedidos y reservas.
- **Diseñar una base de datos relacional** normalizada que garantice integridad referencial.

- **Crear una interfaz gráfica intuitiva** que facilite la operación del sistema por personal con distintos niveles técnicos.
- **Implementar un sistema de reportes** que proporcione información estratégica en tiempo real.
- **Desarrollar un sistema de fidelización** de clientes basado en puntos acumulables.

### **3. RECOLECCIÓN DE INFORMACIÓN**

Para definir los requerimientos del sistema, se aplicaron tres técnicas principales de recolección de información: entrevistas, formularios y observación directa. Aunque inicialmente se buscaron entrevistas con responsables de cafeterías locales, no fue posible concretar reuniones ni recibir respuestas a los formularios enviados.

Ante esta limitación, se optó por realizar observación directa en establecimientos como Typica Café, Ocheto, Comopa, Raza de Bronce y 3600, entre otros. Esta técnica permitió analizar de primera mano los procesos más frecuentes en el funcionamiento de cafeterías reales, tales como toma de pedidos, reservas, control de caja, cobranza, promociones e inventario.

La información recolectada sirvió como base para modelar un sistema de gestión que administre una cafetería, inspirado en las prácticas observadas, y definir los requerimientos funcionales y no funcionales del sistema.

### **4. ALCANCE DEL SISTEMA**

**Incluye:**

- **Gestión de usuarios y roles:** Registro, edición, eliminación, autenticación y control de permisos según el rol asignado (Administrador, Cajero).
- **Administración del menú:** Creación, edición y eliminación de productos, asignación de categorías y actualización de precios o disponibilidad.
- **Gestión de reservas:** Registro de reservas, validación de disponibilidad de mesas, consulta de reservas activas y administración de estados.
- **Gestión de pedidos:** Toma de pedidos, incorporación de múltiples productos y actualización de estados.

- **Cobranza y control de caja:** Generación de recibos con número único, registro de pagos, cálculo de cambio, actualización de caja.
- **Sistema de ofertas y promociones:** Creación, activación y modificación de descuentos aplicables al menú y pedidos.
- **Reportes estadísticos:** Visualización de ventas por período, productos más vendidos, clientes frecuentes y métricas clave del negocio.
- **Sistema de fidelización:** Acumulación de puntos por compra.

**No incluye:**

- Integración con sistemas de pago en línea (pasarelas de pago o billeteras digitales).
- Aplicación móvil para Android o iOS.
- Sistema de delivery con seguimiento por GPS.
- Facturación electrónica con impuestos oficiales según normativas nacionales.
- Módulos avanzados de análisis predictivo o inteligencia artificial.

## 5. ANÁLISIS DE REQUERIMIENTOS

### 5.1 Identificación de Actores

A continuación, se describen los actores que interactúan con el sistema, sus características y sus principales responsabilidades.

#### Actor 1: Cliente

##### **Descripción:**

Usuario externo que realiza reservas, acumula puntos de fidelización y consulta información general del menú. Su interacción con el sistema se realiza a través del Cajero o mediante módulos habilitados específicamente para consulta.

**Responsabilidades:**

- Realizar reservas de mesas.
- Consultar el menú disponible.
- Ver ofertas vigentes.
- Revisar sus reservas activas.
- Acumular puntos de fidelidad al realizar compras.

**Actor 2: Cajero****Descripción:**

Empleado que atiende a los clientes en el establecimiento, gestionando pedidos y realizando las operaciones de facturación y control de caja.

**Responsabilidades:**

- Registrar pedidos y asignar mesas.
- Consultar disponibilidad de platos.
- Actualizar el estado de los pedidos.
- Generar recibos y registrar pagos.
- Gestionar apertura y cierre de caja.
- Ver reservas registradas y confirmarlas cuando corresponda.

**Actor 3: Administrador****Descripción:**

Usuario con máximo nivel de permisos, responsable de la configuración, mantenimiento y supervisión del sistema.

**Responsabilidades:**

- Gestionar usuarios y roles del sistema.
- Administrar categorías, menú y promociones.
- Gestionar inventario e ingredientes.

- Configurar parámetros del sistema.
- Generar reportes avanzados de ventas y actividad operativa.
- Realizar mantenimiento y copias de seguridad de la base de datos.

## 5.2 Casos de Uso Principales

### CU-01: Iniciar Sesión

**Actor principal:** Todos los usuarios

**Precondición:** Usuario registrado en el sistema

**Flujo principal:**

1. Usuario ingresa nombre de usuario y contraseña
2. Sistema valida credenciales
3. Sistema verifica rol del usuario
4. Sistema registra acceso
5. Sistema muestra menú según rol

**Flujo alternativo:**

- 2a. Credenciales incorrectas: mostrar error, permitir 5 intentos.
- 2b. Usuario bloqueado: mostrar mensaje, denegar acceso.

### CU-02: Tomar Pedido

**Actor principal:** Cajero

**Precondición:** Cliente presente en mesa asignada

**Flujo principal:**

1. Cajero selecciona mesa
2. Cajero selecciona cliente
3. Sistema muestra menú disponible
4. Cajero agrega productos al pedido

5. Cajero ingresa notas especiales por plato
6. Sistema calcula subtotal
7. Sistema registra pedido con estado "Pendiente"

**Flujo alternativo:**

- Productos no disponibles: sistema muestra mensaje
- Cliente solicita descuento: aplicar oferta vigente

**CU-03: Generar Recibo**

**Actor principal:** Cajero

**Precondición:** Pedido en estado "Servido"

**Flujo principal:**

1. Cajero selecciona pedido
2. Sistema calcula subtotal, impuestos y total
3. Cajero selecciona método de pago
4. Cajero ingresa monto pagado (si es efectivo)
5. Sistema calcula cambio
6. Sistema actualiza puntos de fidelidad del cliente
7. Sistema genera recibo con número único
8. Sistema actualiza estado de pedido a "Pagado"
9. Sistema libera mesa

**Postcondición:** Recibo registrado, caja actualizada

**CU-04: Generar Reporte de Ventas**

**Actor principal:** Administrador

**Precondición:** Datos de ventas en período solicitado

### **Flujo principal:**

1. Usuario selecciona tipo de reporte
2. Usuario selecciona rango de fechas
3. Sistema consulta base de datos
4. Sistema procesa y agrupa información
5. Sistema genera reporte formateado
6. Sistema muestra estadísticas (totales, promedios)

### **5.3 Lista de Requisitos Funcionales**

#### **RF-01: Autenticación y Control de Acceso**

- **RF-01.01** El sistema debe permitir el inicio de sesión mediante usuario y contraseña.
- **RF-01.02** El sistema debe validar credenciales antes de permitir el acceso.
- **RF-01.03** El sistema debe restringir funcionalidades según el rol del usuario.

#### **RF-02: Gestión del Menú**

- **RF-02.01** El sistema debe permitir consultar el menú completo.
- **RF-02.02** El Administrador debe poder agregar nuevos productos.
- **RF-02.03** El Administrador debe poder editar precios, nombres o disponibilidad.
- **RF-02.04** El Administrador debe poder eliminar productos.
- **RF-02.05** El sistema debe permitir gestionar categorías del menú.

#### **RF-03: Gestión de Reservas**

- **RF-03.01** El sistema debe permitir registrar reservas de mesas.
- **RF-03.02** El sistema debe validar disponibilidad antes de registrar una reserva.
- **RF-03.03** El Cajero debe poder consultar y confirmar reservas.
- **RF-03.04** El sistema debe permitir cancelar reservas.

## **RF-04: Gestión de Pedidos**

- **RF-04.01** El sistema debe permitir añadir múltiples productos a un pedido.
- **RF-04.02** El sistema debe actualizar el estado del pedido (“Pendiente”, “En preparación”, “Servido”).
- **RF-04.03** El sistema debe calcular el subtotal del pedido automáticamente.

## **RF-05: Cobranza y Caja**

- **RF-05.01** El sistema debe generar recibos con número único.
- **RF-05.02** El sistema debe registrar el método de pago.
- **RF-05.03** El sistema debe calcular cambio en pagos en efectivo.
- **RF-05.04** El sistema debe actualizar el estado del pedido a “Pagado”.
- **RF-05.05** El sistema debe registrar movimientos de caja.

## **RF-06: Fidelización**

- **RF-06.01** El sistema debe asignar puntos por cada compra realizada.
- **RF-06.02** El sistema debe permitir canjear puntos según reglas definidas.

## **RF-07: Ofertas y Promociones**

- **RF-07.01** El Administrador debe poder crear ofertas o descuentos.
- **RF-07.02** El sistema debe aplicar la oferta si corresponde al pedido.
- **RF-07.03** El Cajero debe poder consultar promociones vigentes.

## **RF-08: Inventario**

- **RF-08.01** El Administrador debe poder registrar ingredientes o insumos.
- **RF-08.02** El sistema debe permitir actualizar cantidades de stock.
- **RF-08.03** El sistema debe generar alertas de stock bajo.

## **RF-9: Reportes**

- **RF-9.01** El Administrador debe poder generar reportes de ventas por período.

- **RF-9.02** El sistema debe permitir consultar productos más vendidos.
- **RF-9.03** El sistema debe generar estadísticas de clientes frecuentes.
- **RF-9.04** El sistema debe mostrar métricas resumidas y totales

## 6. PLANIFICACIÓN Y GESTIÓN DEL PROYECTO

### 6.1 Cronograma de Desarrollo

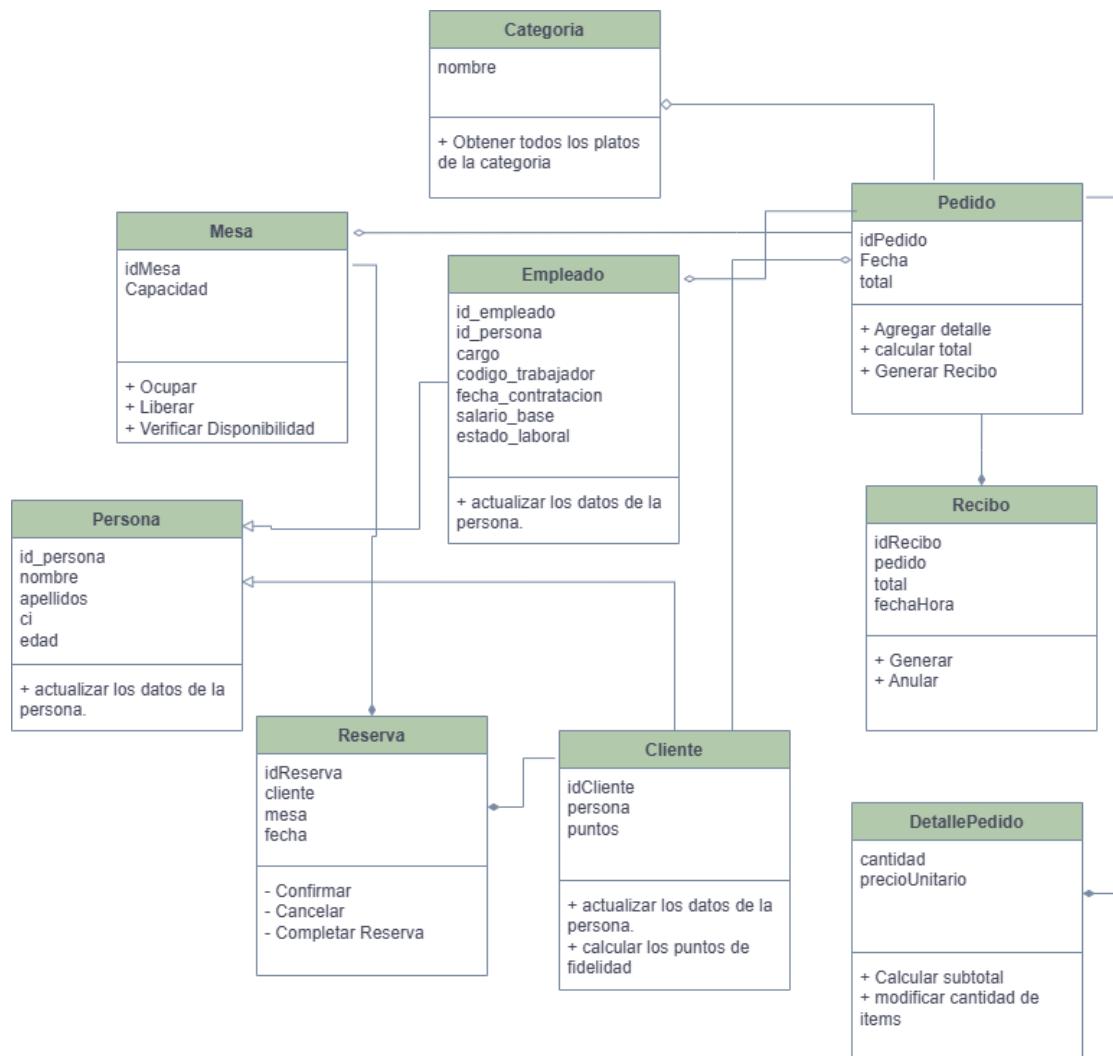
Actividades	Septiembre				Octubre				Noviembre				Diciembre				
	1	2	3	4	1	2	3	4	1	2	3	4	1	1	2	2	
Planteamiento del Problema	■	■															
Revisión teórica sobre una cafetería y recolección de datos			■	■													
Diseño conceptual de la base de datos					■												
Diseño lógico y normalización					■												
Creación de diagrama UML						■	■										
Configuración de conexión							■	■									
Diseño de la interfaz gráfica y programación en JAVA							■	■	■	■	■						
Redacción del informe final												■	■	■	■		
Entrega y presentación del proyecto																■	

## 6.2 Asignación de Roles

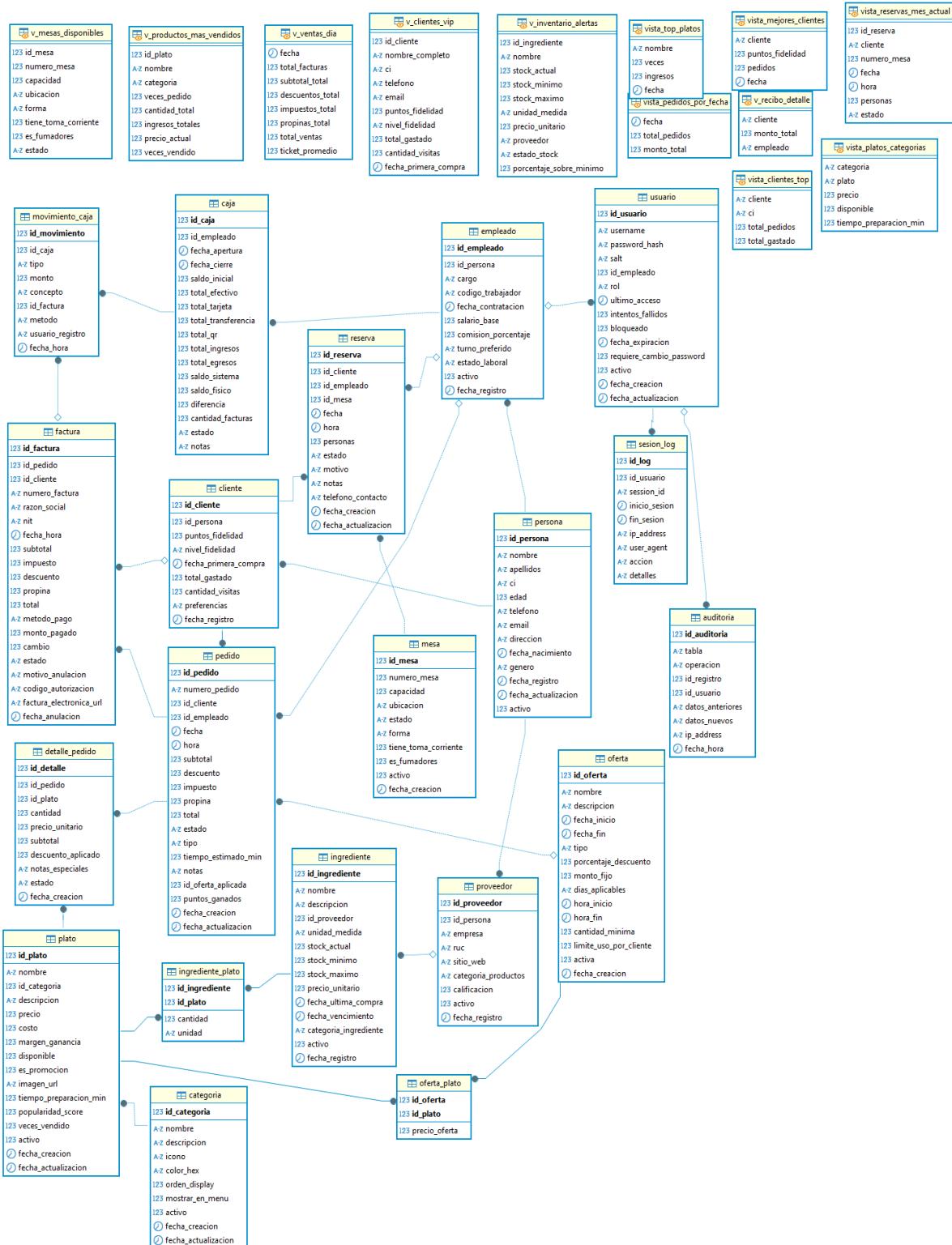
Rol	Responsable	Responsabilidades principales
Project Manager	Gonzalo Apaza	Coordinación general, seguimiento
Analista de BD	Gonzalo Apaza	Diseño y optimización de BD
Backend Developer	Gonzalo Apaza	Capa de datos y lógica de negocio
Frontend Developer	Gonzalo Apaza	Interfaces gráficas y UX
QA Tester	Gonzalo Apaza	Pruebas y control de calidad

## 7. MODELADO DEL SISTEMA

### 7.1 Diagrama de Clases (UML)



## 7.2 Diagrama Entidad - Relación



## **8. DESCRIPCIÓN DE LA ARQUITECTURA**

### **8.1 Arquitectura en Tres Capas**

El sistema de gestión para cafetería/restaurante implementa una arquitectura multicapa cuyo objetivo es separar claramente las responsabilidades, mejorar la mantenibilidad, la escalabilidad y facilitar la evolución futura del sistema. Las tres capas principales son:

### **8.2 Capa de Presentación (PRESENTACIÓN)**

#### **Responsabilidades:**

- Mostrar información al usuario
- Capturar entrada del usuario
- Validar formato de datos
- Invocar servicios de la capa de negocio
- Actualizar la interfaz según respuestas

#### **Componentes principales:**

- **frmLogin:** Pantalla de autenticación
- **frmMenuPrincipal:** Menú principal según el rol del usuario
- **frmMenuCliente:** Portal del cliente
- **frmMenuPuntoVenta:** Interfaz para punto de venta
- **frmMenuAdministracion:** Panel administrativo
- **frmCategoria:** Gestión de categorías
- **frmCliente:** Gestión de clientes
- **frmDashboard:** Visualización de estadísticas e indicadores
- **frmGestionUsuarios:** Gestión de usuarios, solo disponible para administrador
- **frmFactura:** Generación de recibo.
- **frmFidelidad:** Gestión del programa de fidelización

- **frmInventario:** Administración del inventario
- **frmOfertas:** Gestión de ofertas y promociones
- **frmPlato:** Gestión de platos o productos
- **frmReportes:** Visualización de reportes
- **frmReserva:** Gestión de reservas
- **frmVenta:** Registro y control de ventas
- **UIConstants:** Constantes de diseño visual

**Tecnología:** Java Swing (JFrame, JPanel, JButton, JTable, etc.)

### 8.3 Capa de Lógica de Negocio (LOGICA)

#### Responsabilidades:

- Implementar reglas de negocio
- Coordinar operaciones entre objetos
- Validar consistencia de datos
- Aplicar restricciones del dominio

#### Componentes principales:

- Usuario: Gestión de autenticación y roles
- Cliente: Lógica de fidelización
- Empleado: Información de personal
- Plato: Disponibilidad y precios
- Pedido: Cálculo de totales, aplicación de ofertas
- Reserva: Validación de disponibilidad
- Recibo: Generación de total de la cuenta.
- Sesión: Gestión de sesión activa

### **Principios aplicados:**

- Encapsulamiento de lógica de negocio
- Independencia de la capa de presentación
- Independencia de la base de datos

### **8.4 Capa de Persistencia (DATOS)**

#### **Responsabilidades:**

- Gestionar conexión a base de datos
- Ejecutar consultas SQL
- Mapear resultados a objetos
- Manejar transacciones

#### **Componentes principales:**

- conexion: Gestión de conexión a MySQL
- usuarioDT: CRUD de usuarios
- clienteDT: CRUD de clientes
- platosDT: CRUD de productos
- reservaDT: CRUD de reservas
- pedidoDT: CRUD de pedidos
- facturaDT: CRUD de recibos
- mesaDT: CRUD de mesas
- ingredienteDT: CRUD de ingredientes

**Patrón aplicado:** DAO (Data Access Object)

## **9. DESARROLLO TECNOLÓGICO**

El sistema fue desarrollado utilizando un conjunto de tecnologías complementarias que permiten garantizar la eficiencia, estabilidad y escalabilidad del software. La elección de cada herramienta se realizó considerando criterios como disponibilidad, rendimiento, facilidad de mantenimiento, compatibilidad y curva de aprendizaje.

### **9.1 Tecnologías Utilizadas**

A continuación, se presenta el conjunto de tecnologías implementadas en cada capa del sistema:

<b>Capa del Sistema</b>	<b>Tecnología / Herramienta</b>	<b>Justificación</b>
<b>Presentación (GUI)</b>	Java Swing, JFrame, JPanel, JTable, UIConstants	Permiten diseñar interfaces gráficas robustas, personalizables y multiplataforma.
<b>Lógica de Negocio</b>	Java	Facilita la aplicación de principios OOP, separación de lógica y mantenimiento del código.
<b>Persistencia de Datos</b>	MySQL	Motor de base de datos eficiente, confiable y ampliamente usado con JDBC.
<b>Acceso a Datos</b>	JDBC (Java Database Connectivity)	Conexión directa y controlada a la base de datos para consultas y transacciones.
<b>Control de Versiones</b>	Git y GitHub	Permite mantener historial de cambios, control de versiones y respaldo del proyecto.
<b>Modelado UML</b>	Draw.io	Herramienta simple y eficiente para crear diagramas UML y estructurar el sistema.

## **10. MANUAL DE USUARIO**

### **10.1 Introducción**

El Sistema de Administración de Cafetería es una herramienta integral diseñada para gestionar las operaciones principales del negocio.

Incluye diferentes módulos según el tipo de usuario:

- Administración
- Punto de Venta
- Reservas de Mesa

Cada módulo permite realizar tareas específicas como gestión de productos, registro de ventas, administración de usuarios, generación de reportes y control de reservas.

Este manual explica cómo navegar y utilizar cada uno de los módulos mostrados en el sistema.

### **10.2 Inicio del Sistema**

1. Ejecute el programa desde su acceso directo o archivo ejecutable.
2. Se mostrará la ventana de inicio de sesión.
3. Ingrese sus credenciales:
  - Usuario
  - Contraseña
4. Haga clic en “Iniciar sesión”.

Dependiendo del rol del usuario (Administrador o Cajero), aparecerán diferentes módulos habilitados.

### **10.3 Panel Principal (Selección de Módulos)**

Una vez iniciada la sesión, se muestran tres módulos principales:

- Administración
- Punto de Venta

- Reservas

El usuario debe seleccionar el módulo con el que desea trabajar.

El acceso a algunos módulos está restringido según el rol asignado.

#### **10.4 Módulo de Administración**

Este módulo está disponible solo para Administradores.

Desde aquí se puede acceder a:

- Gestión de Productos
- Gestión de Categorías
- Promociones y Ofertas
- Reportes
- Gestión de Usuarios
- Configuración (incluye Respaldo de Base de Datos)

##### **Gestión de Productos**

Permite agregar, editar o eliminar productos del menú.

Funciones disponibles:

- Registrar producto nuevo
- Actualizar información
- Eliminar productos
- Buscar productos existentes

##### **Gestión de Categorías**

Permite administrar las categorías del menú:

- Agregar nuevas categorías
- Editar categorías existentes
- Eliminar categorías que ya no se utilicen

## **Promociones y Ofertas**

En esta ventana puede:

- Crear una nueva oferta o promoción
- Eliminar promociones activas
- Actualizar información de ofertas

## **Reportes (Administración)**

El sistema incluye un módulo de reportes con filtros por fecha.

Se pueden obtener reportes de ventas, productos, clientes u otros datos relevantes de la cafetería.

Aparece un mensaje automático cuando no existen ingredientes en bajo stock.

## **Dashboard**

Ventana que resume la información más importante del negocio:

- Productos más vendidos
- Top clientes
- Ventas del día
- Cantidad de reservas
- Pedidos realizados en el día

## **Gestión de Usuarios**

Disponible únicamente para administradores.

Se permite:

- Ver lista de usuarios
- Crear nuevos usuarios
- Cambiar contraseñas
- Eliminar usuarios

## **Respaldo de Base de Datos**

Permite generar un respaldo del sistema en cualquier carpeta del dispositivo.

## **10.5 Punto de Venta**

Este módulo está disponible para administradores y cajeros.

Opciones principales:

- Nueva Venta
- Recibos
- Clientes
- Reportes

### **Nueva Venta**

Aquí se pueden realizar ventas y cobro:

- Registrar cliente
- Seleccionar productos
- Modificar o eliminar ítems del pedido
- Aplicar métodos de pago
- Generar recibo

### **Sistema de Cobranza**

Se muestra el recibo generado y los detalles del cobro realizado.

### **Gestión de Clientes**

Permite:

- Registrar nuevos clientes
- Ver información de clientes frecuentes

## **Reportes**

Similar al módulo de administración, permite generar reportes de ventas y movimientos realizados desde el punto de venta.

## **10.6 Módulo de Reservas de Mesa**

Permite administrar las reservas activas y crear nuevas.

### **Funciones:**

- Ver lista de reservas
- Consultar fecha, hora, mesa y cantidad de personas
- Cancelar reservas
- Crear nuevas reservas
- Registrar cliente nuevo
- Confirmar reserva seleccionando fecha y hora

## **10.7 Búsquedas y Filtros**

En varios módulos se incluyen barras de búsqueda para encontrar:

- Productos
- Clientes
- Ventas
- Usuarios
- Reservas

Solo ingrese parte del nombre o dato y presione Enter o el ícono de búsqueda.

## **10.8 Cerrar Sesión o Salir**

Para cerrar sesión:

- Presione Cerrar sesión en el menú superior o módulo principal.

Para salir del sistema:

- Haga clic en Salir
- O cierre la ventana del programa.

Es recomendable cerrar sesión correctamente para proteger los datos del sistema.

## **11. CONCLUSIONES Y APRENDIZAJES**

### **11.1 Conclusiones**

La implementación de un sistema de administración mejora la organización interna de la cafetería, permitiendo un mayor control bajo una arquitectura modular que facilita el mantenimiento de las actividades operativas como inventario, ventas, pedidos, reservas y cobranzas.

Se evidenció que la digitalización de procesos reduce errores humanos relacionados con el registro de pedidos y el cálculo de pagos, evitando pérdidas económicas y aumentando la precisión en la gestión de caja.

El sistema facilitó un control más eficiente del inventario, permitiendo prever necesidades de abastecimiento y evitar la falta de productos.

A través de los reportes de ventas generados por el sistema, se logró tomar decisiones informadas, identificando productos más vendidos y la fidelidad del cliente.

La automatización de procesos administrativos contribuyó a optimizar el tiempo del personal, enfocándose más en la atención al cliente y en mejorar la experiencia del cliente.

### **11.2 Aprendizajes**

Se comprendió la importancia de integrar las áreas de ventas, inventario, reservas y cobranza en un solo sistema para garantizar un flujo de información coherente y actualizado.

Se aprendió que una interfaz sencilla y amigable es fundamental para que el personal adopte el sistema sin dificultades ni resistencia al cambio.

El proceso mostró que el análisis de datos es fundamental para el desarrollo estratégico del cual permite proyectar el crecimiento modular, optimizar el menú y ajustar precios.

Se identificó que la seguridad de la información como accesos restringidos, respaldos y protección de datos es un componente esencial para evitar riesgos.

## 12. ANEXOS

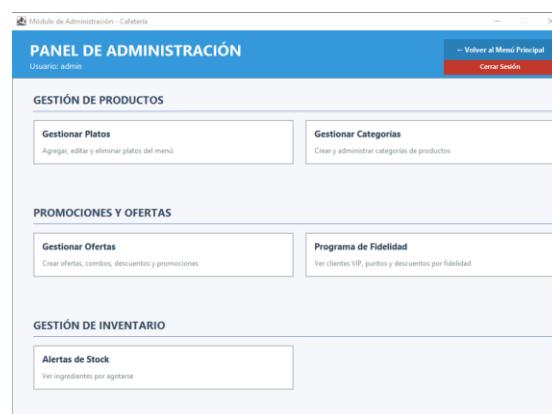
### 12.1 Capturas GUI



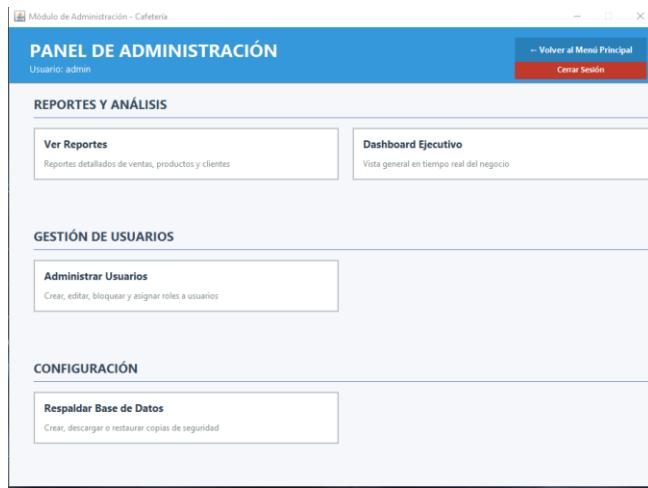
Pantalla de autenticación.



Menú de opciones a los tres módulos: Administración, Punto de venta y Reservas de mesa.



Módulo de administración 1: presenta las opciones para la gestión de productos, promociones, ofertas y gestión de inventario.



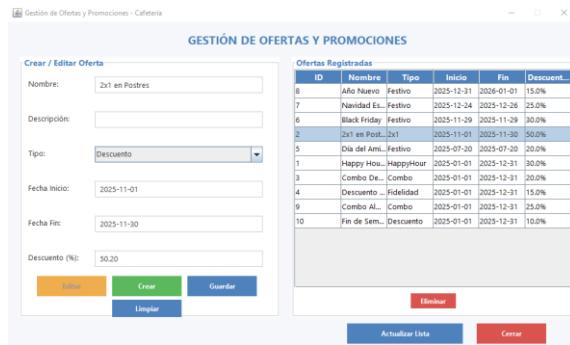
Módulo de administración 2: presenta las opciones para ver reportes, gestión de usuarios y configuración.



Gestión de productos: Donde se puede agregar, editar o eliminar el producto.



Gestión de categorías: Donde se puede eliminar, agregar, actualizar las categorías de los productos.



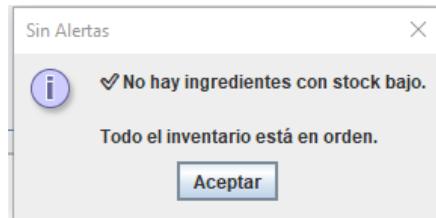
Gestión de ofertas y promociones: Donde se puede crear una nueva oferta o promoción, así mismo se puede modificar y eliminar alguna de estas ofertas.



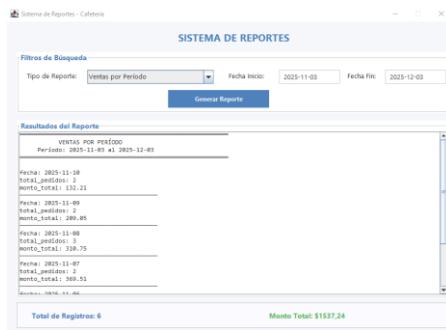
Programa de fidelidad: Donde se muestra la lista de los clientes más fieles.



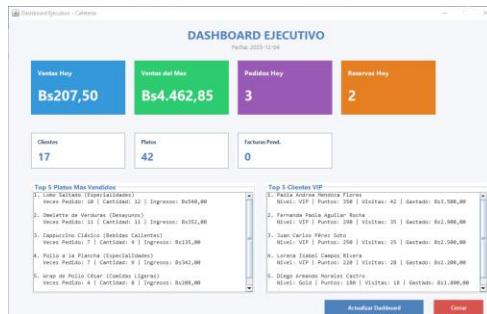
La ventana permite gestionar usuarios del sistema. Incluye un campo de búsqueda con botón Filtrar y botones inferiores para cambiar contraseña, eliminar usuarios y cerrar la ventana.



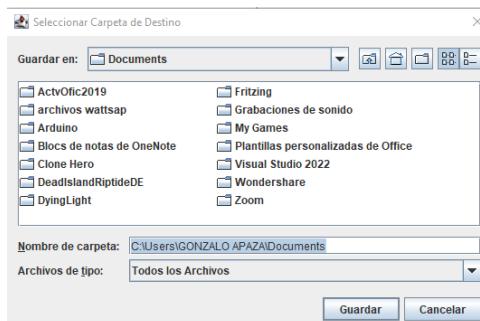
Alerta: Muestra un mensaje que se cuenta con el stock de ingredientes suficiente.



### Módulo de reportes generales.



Ventana del Dashboard, muestra los detalles más importantes de la cafetería, como el top clientes, los productos más vendidos, las ventas del día, cantidad de reservas, cantidad de pedidos.



Respaldo de la base de datos: Con esta opción se puede guardar un respaldo en cualquier carpeta del dispositivo en el que se ejecuta este programa.



Menú del Punto De Venta: Presenta cuatro opciones para el cajero y son el modulo de Nueva Venta, Recibos, Clientes y Reportes.

Pantalla de registro de ventas: Registra al cliente, su orden y formas de pago generando un recibo. También puede modificar o eliminar el pedido.

Ventana de sistema de cobranza: Muestra recibo y los detalles del mismo.

**GESTIÓN DE CLIENTES**

**Datos del Cliente**

Nombre:

Apellidos:

Cl:

Edad:

**Registrar** **Limpiar**

**Lista de Clientes**

- ID: 1 | Juan Carlos Pérez Soto  
CI: 1234567LP | Edad: 30 años
- ID: 2 | María Elena López García  
CI: 2345678CB | Edad: 25 años
- ID: 3 | Paola Andrea Mendoza Flores  
CI: 6789012CB | Edad: 27 años
- ID: 4 | Roberto Miguel Sánchez Cordero  
CI: 9012345SC | Edad: 33 años
- ID: 5 | Carmen Rosa Flores Quispe  
CI: 0123456LP | Edad: 26 años
- ID: 6 | Diego Armando Morales Castro  
CI: 1122334LP | Edad: 31 años
- ID: 15 | Valeria Beatriz Herrera Pinto  
CI: 3333333LP | Edad: 24 años

**Actualizar Lista** **Cerrar**

Registro de clientes: Se puede registrar un nuevo cliente.

**SISTEMA DE REPORTES**

**Filtros de Búsqueda**

Tipo de Reporte: **Ventas por Período** Fecha Inicio: 2025-11-03 Fecha Fin: 2025-12-03

**Generar Reporte**

**Resultados del Reporte**

VENTAS POR PERÍODO		
Período: 2025-11-03 al 2025-12-03		
fecha:	2025-11-10	total_pedidos: 2
		monito_total: 332,21
fecha:	2025-11-09	total_pedidos: 2
		monito_total: 289,05
fecha:	2025-11-08	total_pedidos: 3
		monito_total: 310,75
fecha:	2025-11-07	total_pedidos: 2
		monito_total: 369,51
fecha:	2025-11-06	total_pedidos: 1
		monito_total: 10,00

Total de Registros: 6 Monto Total: \$1537,24

Ventana de reportes: Muestra reportes de ventas.

**GESTOR DE RESERVAS**

Sistema de Administración de Cafetería Usuario: admin | Rol: Administrador

Reservas Activas: 19 | Pendientes: 0 | Confirmadas: 0

ID	Mesa	Personas	Fecha	Hora	Acciones
1	2	4	2025-11-10	19:00	Ver detalles
2	1	2	2025-11-10	13:00	Ver detalles
3	5	8	2025-11-11	20:00	Ver detalles
4	3	4	2025-11-11	14:30	Ver detalles
5	4	6	2025-11-12	19:30	Ver detalles
6	7	4	2025-11-12	12:00	Ver detalles
7	8	6	2025-11-13	21:00	Ver detalles
8	11	10	2025-11-13	18:00	Ver detalles
9	2	4	2025-11-14	13:30	Ver detalles
10	6	2	2025-11-14	15:00	Ver detalles

**Nueva Reserva** **Editar** **Cancelar** **Volver**

Selecciona una reserva de la tabla para editarla o cancelarla. Haz clic en 'Nueva Reserva' para crear una nueva.

Ventana de gestor de reservas: Administración de reservas permite editar y cancelar reservas.



Ventana para nuevas reservas: Permite nueva reserva asigna mesa, fecha, hora y cantidad de personas.

## 12.2 Capturas del código

```

CREATE OR REPLACE VIEW v_productos_mas_vendidos AS
SELECT
    p.id_plato
    , p.nombre
    , c.nombre AS categoria
    , COUNT(dp.id_detalle)      AS veces_pedido
    , SUM(dp.cantidad)         AS cantidad_total
    , SUM(dp.subtotal)          AS ingresos_totales
    , p.precio                  AS precio_actual
    , p.veces_vendido
FROM detalle_pedido dp
INNER JOIN plato p
    ON dp.id_plato = p.id_plato
INNER JOIN categoria c
    ON p.id_categoria = c.id_categoria
INNER JOIN pedido ped
    ON dp.id_pedido = ped.id_pedido
WHERE ped.estado IN ('Pagado', 'Servido')
GROUP BY
    p.id_plato
    , p.nombre
    , c.nombre
    , p.precio
    , p.veces_vendido
ORDER BY cantidad_total DESC;

```

La vista de productos más vendidos

```

1 package DATOS;
2 import java.sql.*;
3
4 public class conexion {
5
6     private static final String URL = "jdbc:mysql://localhost:3306/dbcafeteria?useSSL=false&serverTimezone=UTC";
7     private static final String USER = "root";
8     private static final String PASSWORD = "";
9
10    public static Connection conectar() {
11        Connection conexion = null;
12        try {
13            Class.forName("com.mysql.cj.jdbc.Driver");
14            conexion = DriverManager.getConnection(URL, USER, PASSWORD);
15            System.out.println("Conexión establecida con éxito a MySQL.");
16        } catch (ClassNotFoundException e) {
17            System.err.println("Error: Clase no encontrada.");
18        } catch (SQLException e) {
19            System.err.println("Error al conectar a la base de datos: " + e.getMessage());
20        }
21        return conexion;
22    }
23
24    public static void desconectar(Connection conexion) {
25        if (conexion != null) {
26            try {
27                conexion.close();
28                System.out.println("Conexión cerrada correctamente.");
29            } catch (SQLException e) {
30                System.err.println("Error al cerrar la conexión: " + e.getMessage());
31            }
32        }
33    }

```

Código en JAVA para la conexión de eclipse con la base de datos.

```

211     StringBuilder sbPlatos = new StringBuilder();
212     String sqlTopPlatos = "SELECT * FROM v_productos_mas_vendidos ORDER BY cantidad_total DESC LIMIT 5";
213
214     try (PreparedStatement ps = conn.prepareStatement(sqlTopPlatos)) {
215         ResultSet rs = ps.executeQuery();
216         int pos = 1;
217         while (rs.next()) {
218             sbPlatos.append(String.format("%d. %s (%s)\n",
219                 pos++,
220                 rs.getString("nombre"),
221                 rs.getString("categoria")));
222         }
223         sbPlatos.append(String.format(
224             "Veces Pedido: %d | Cantidad: %d | Ingresos: %s\n\n",
225             rs.getInt("veces_pedido"),
226             rs.getInt("cantidad_total"),
227             formatoMoneda.format(rs.getDouble("ingresos_totales")));
228     });
229     txaTopPlatos.setText(sbPlatos.toString());
230
231     StringBuilder sbClientes = new StringBuilder();
232     String sqlTopClientes = "SELECT * FROM v_clientes_vip ORDER BY puntos_fidelidad DESC LIMIT 5";
233
234     try (PreparedStatement ps = conn.prepareStatement(sqlTopClientes)) {
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254

```

Código en la que se utilizan las vistas de la base de datos para mostrar la información.

```

227     private void respaldarBaseDeDatos() {
228         JFileChooser fileChooser = new JFileChooser();
229         fileChooser.setDialogTitle("Seleccionar Carpeta de Destino");
230         fileChooser.setFileSelectionMode(JFileChooser.DIRECTORIES_ONLY);
231
232         int userSelection = fileChooser.showSaveDialog(this);
233         if (userSelection == JFileChooser.APPROVE_OPTION) {
234             File carpeta = fileChooser.getSelectedFile();
235
236             try (Connection conn = conexion.conectar()) {
237
238                 if (conn == null) {
239                     JOptionPane.showMessageDialog(this,
240                         "No se pudo conectar a la base de datos.",
241                         "Error", JOptionPane.ERROR_MESSAGE);
242                     return;
243                 }
244
245                 String db = conn.getCatalog();
246                 if (db == null) {
247                     JOptionPane.showMessageDialog(this,
248                         "No se pudo obtener el nombre de la base de datos.",
249                         "Error", JOptionPane.ERROR_MESSAGE);
250                     return;
251                 }
252
253                 String user = "root";
254                 String password = "";

```

Código para generar un respaldo de la base de datos.

```
UIConstants.java X
1 package PRESENTACION;
2
3 import java.awt.*;
4
5 public class UIConstants {
6     public static final Color COLOR_FONDO = new Color(245, 247, 250);           // Azul azulado muy clara
7     public static final Color COLOR_PANEL = new Color(255, 255, 255);           // Blanca
8     public static final Color COLOR_PRIMARY = new Color(79, 129, 189);          // Azul oscuro
9     public static final Color COLOR_SUCCESS = new Color(92, 164, 92);           // Verde oscuro
10    public static final Color COLOR_WARNING = new Color(240, 178, 78);          // Naranja oscuro
11    public static final Color COLOR_DANGER = new Color(217, 83, 79);            // Roja oscuro
12    public static final Color COLOR_TEXTO = new Color(51, 51, 51);              // Gris oscuro
13    public static final Color COLOR_TEXTO_CLARO = new Color(255, 255, 255);        // Blanca
14
15    public static final Font FUENTE_TITULO = new Font("Segoe UI", Font.BOLD, 20);
16    public static final Font FUENTE_SUBTITULO = new Font("Segoe UI", Font.BOLD, 14);
17    public static final Font FUENTE_NORMAL = new Font("Segoe UI", Font.PLAIN, 13);
18    public static final Font FUENTE_BOTON = new Font("Segoe UI", Font.BOLD, 12);
19
20●   public static void estilizarBoton(javax.swing.JButton boton, Color colorFondo) {
21       boton.setFont(FUENTE_BOTON);
22       boton.setBackground(colorFondo);
23       boton.setForeground(COLOR_TEXTO_CLARO);
24       boton.setFocusPainted(false);
25       boton.setBorderPainted(false);
26       boton.setCursor(new Cursor(Cursor.HAND_CURSOR));
27   }
28 }
29
```

UIConstants.java son estilos de la interfaz, definiendo colores y tipografías de acuerdo a su diseño.

Incluye estilizarBoton() para aplicar fuente, colores, borde y cursor de forma uniforme.

### 13. LINK DEL REPOSITORIO

[https://github.com/gonzaloapaza-creator/SG\\_CAFETERIA\\_.git](https://github.com/gonzaloapaza-creator/SG_CAFETERIA_.git)