



Práctica unidad 1 Programación Avanzada

BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

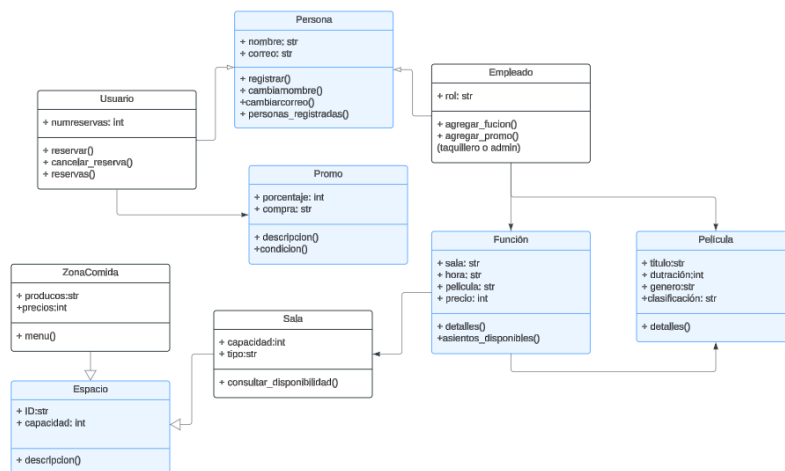
MARTÍNEZ RUGERIO LUCÍA

PROGRAMACIÓN AVANZADA

M.C. JAIME ALEJANDRO ROMERO SIERRA

12 DE FEBRERO DE 2025

1.- Sistema de Reservas para un Cine



- **Justificación del programa:**

Este programa sirve para generar reservas, y tener una mejor administración de los clientes y las salas con funciones disponibles. De esta forma no se duplican asientos o se sobrevenden boletos.

- **Clases:**

Película:

- Atributos: título, duración, género, clasificación
- Métodos: detalles, muestra los detalles de la película

Funcion:

- Atributos: sala, hora, película, precio
- Métodos: detalles, asientos disponibles

Promo:

- Atributos: porcentaje, compra
- Métodos: descripción, condición

Espacio

- Atributos: ID, capacidad
- Métodos: descripción

ZonaComida

- Atributos: productos, precios
- Métodos: menú

Sala

- Atributos: tipo
- Métodos: consultar disponibilidad

Persona

- Atributos: nombre, correo
- Métodos: registrar, cambiar nombre, cambiar correo, personas registradas

Usuario

- Atributos:
- Métodos: reservar, cancelar reserva, reservas

Empleado

- Atributos: rol
- Métodos: agregar promo, agregar función

- Código ejecutado:

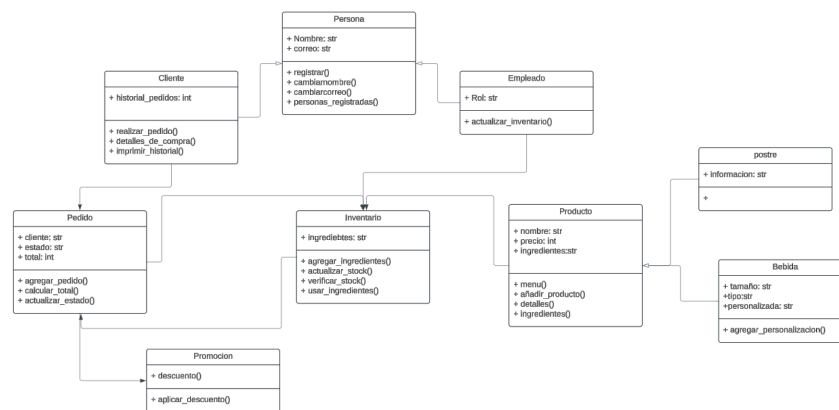
```
Pelicula1= Pelicula("Avatar", 162,"Acción/Ciencia Ficción","PG-13")
Pelicula2= Pelicula("CJ7", 88, "Ciencia ficción humorística/Fantasía y familiar", "PG")
Pelicula1.detalles()
Pelicula2.detalles()
productos1= ["Hot Dog", "Palomitas", "Hamburguesa", "Refrescos", "Agua natural"]
precios1=[35, 20, 120, 15, 10]
Zona1= ZonaComida("Zona 1", 30, productos1, precios1)
Zona1.descripcion()
Zona1.menu()
Sala1= Sala("Sala 1", 100, "2D")
Sala1.descripcion()
Sala1.consultar_disponibilidad()
Sala2= Sala("Sala 2", 50, "3D")
Sala2.descripcion()
Sala2.consultar_disponibilidad()
Sala3= Sala("Sala 3", 70, "2D")
Sala3.descripcion()
Sala4= Sala("Sala 4", 100, "4D")
Sala4.descripcion()
Funcion1=Funcion(Sala1, "14:30", Pelicula2, 75)
Funcion2=Funcion(Sala2, "20:00", Pelicula1, 200)
Funcion1.detalles()
Funcion2.detalles()
Promo1=Promo(30, 2)
Promo1.descripcion()
Promo2=Promo(50, 10)
Promo2.descripcion()
Usuario1= Usuario("María", "maria@gmail.com")
Usuario1.registrar()
Usuario1.reservar(Funcion1, 100, Promo1)
Funcion1.asientos_disponibles()
Usuario2= Usuario("Ximena", "ximena@alm.buap.mx")
Usuario2.registrar()
Usuario2.reservar(Funcion2, 10, Promo2)
```

```
Titulo: Avatar |Duración: 162 min |Género: Acción/Ciencia Ficción |Clasificación: PG-13
Titulo: CJ7 |Duración: 88 min |Género: Ciencia ficción humorística/Fantasía y familiar |Clasificación: PG
El espacio Zona 1, tiene capacidad para 30 personas
Menú:
Hot Dog: $35
Palomitas: $20
Hamburguesa: $120
Refrescos: $15
Agua natural: $10
El espacio Sala 1, tiene capacidad para 100 personas
La Sala 1 está disponible
El espacio Sala 2, tiene capacidad para 50 personas
La Sala 2 está disponible
El espacio Sala 3, tiene capacidad para 70 personas
El espacio Sala 4, tiene capacidad para 100 personas
La función es en la Sala 1 a las 14:30, se proyectará la película CJ7. Costo: $75
La función es en la Sala 2 a las 20:00, se proyectará la película Avatar. Costo: $200
Aplica 30% de descuento al comprar 2 boletos
Aplica 50% de descuento al comprar 10 boletos
Se registró correctamente a María con correo de contacto maria@gmail.com
Se realizó una reserva para CJ7 en la Sala 1
No se aplicó ninguna promoción
El costo es de $7500
Asientos disponibles: 0
Se registró correctamente a Ximena con correo de contacto ximena@alm.buap.mx
...
Reservas de María:
No tienes reservas
La Sala 1 está disponible
Asientos disponibles: 100
```

Output is truncated. View as a [scrollable element](#) or open in a [text editor](#). Adjust cell output [settings](#)...

0

2. Gestión de Pedidos en una Cafetería



- **Justificación del programa:**

Este programa nos ayuda a personalizar las bebidas de cada cliente, aplicar promociones de acuerdo a la compra y hacer pedidos rápidos.

- **Clases:**

Inventario:

-Atributos: diccionario

-Métodos: agregar ingredientes, actualizar stock, verificar stock, usar ingredientes

Producto:

-Atributos: nombre, precio, ingredientes

-Métodos: añadir producto, detalles, ingredientes, menu

Bebida:

-Atributos: tamaño, tipo, personalización

-Métodos: agregar personalización

Postre:

-Atributos: información

-Métodos:

Pedido:

-Atributos: cliente

-Métodos: agregar pedido, calcular total, actualizar estado

Persona:

-Atributos: nombre, correo

-Métodos: registrar, cambiar nombre, cambiar correo, personas registradas

Cliente:

-Atributos: lista historial pedidos

-Métodos: realizar pedido, detalles de compra, imprimir historial

Empleado

-Atributos: rol

-Métodos: actualizar inventario

Promoción:

-Atributos: descuento

- Métodos: aplicar descuento

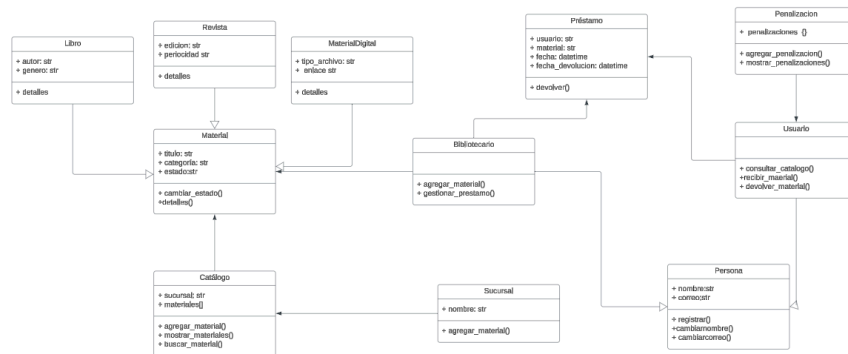
- Código ejecutado

```
inventario= Inventario()
inventario.agregar_ingredient("Leche", 2)
inventario.agregar_ingredient("Cafe", 1)
inventario.agregar_ingredient("Azúcar", 40)
inventario.agregar_ingredient("limón", 3)
Producto1= Bebida("Café con leche", 35, "Mediano", "Frio", "Sin azúcar", {"Cafe":2, "Leche":1})
Producto1.añadir_producto()
Producto2= Postre("Carlota", 150, "No vegano", {"Leche":1, "limón":2, "Azúcar":3})
Producto2.añadir_producto()
Producto.menu()
Empleado1= Empleado("Sofía", "sofia@327.com", "mesera")
Empleado1.registrar()
Empleado1.actualizar_inventario(inventario, "Leche", 2)
Empleado2=Empleado("Luis", "luis@gmial.com", "gerente")
Empleado2.registrar()
Empleado2.actualizar_inventario(inventario, "Leche", 1)
Cliente1= Cliente("Gaby", "gaby@outlook")
Cliente1.registrar()
Persona.personas_registradas()
Pedido1=Pedido(Cliente1)
Pedido2= Pedido(Cliente1)
Pedido1.agregar_pedido(Producto1, 2, inventario)
Pedido1.calcular_total()
Pedido2.agregar_pedido(Producto2, 1, inventario)
Pedido2.calcular_total()
promocion1= Promocion(50)
promocion1.aplicar_descuento(Pedido2)
```

✓ 0.0s

```
¡Nuevo proucto! - Café con leche -
¡Nuevo proucto! - Carlota -
Menú:
-Café con leche --$35
-Carlota --$150
Se registró correctamente a Sofía con correo de contacto sofia@327.com
No tiene permiso para actualizar el inventario
Se registró correctamente a Luis con correo de contacto luis@gmial.com
Stock actualizado: leche - 1
Se ha actualizado el stock
Se registró correctamente a Gaby con correo de contacto gaby@outlook
Personas registradas:
-Sofía - sofia@327.com
-Luis - luis@gmial.com
-Gaby - gaby@outlook
No hay suficiente stock para el producto: Café con leche
Pedido agregado 1- Carlota
Se ha aplicado un descuento de 50%
Total del pedido: 75.00
```

3. Biblioteca Digital



- **Justificación del programa:**

Este programa nos ayuda a agregar libros, revistas y materiales digitales, moverlos entre sucursales, y ayudar al cliente a encontrar un libro o saber si está disponible o no.

- **Clases:**

Material:

-Atributos: título, categoría

-Métodos: cambiar estado, detalles

Libro:

-Atributos: autor, género

-Métodos: detalles

Revista:

-Atributos: edición, periodicidad

-Métodos: detalles

MaterialDigital:

-Atributos: tipo archivo, enlace

-Métodos: detalles

Persona:

-Atributos: nombre, correo

-Métodos: registrar, cambiar nombre, cambiar correo, personas registradas

Usuario:

-Atributos: materiales prestados[]

-Métodos: consultar catálogo, recibir material, devolver material

Bibliotecario:

-Atributos:

-Métodos: agregar material, gestionar préstamo

Sucursal:

-Atributos: nombre

-Métodos: agregar material

Préstamo:

-Atributos: usuario, material, fecha de préstamo, días de devolución

-Métodos: devolver

Penalización:

-Atributos: monto

-Métodos: agregar penalización, mostrar penalización

Catálogo:

-Atributos: sucursal, materiales[]

-Métodos: agregar material, mostrar materiales, buscar material

- Código ejecutado

```
Usuario1= Usuario("Fernanda", "fer123@buap.mx")
Usuario1.registrar()
Bibliotecario1= Bibliotecario("Carlos", "carlos321@buap.mx")
Bibliotecario1.registrar()
Libro1= Libro("El lorax", "Dr. Seus", "Infantil")
Revista1= Revista("Chismes", "II", "Mensual")
MaterialDigital1= MaterialDigital("Libro de Álgebra Lineal", "PDF", "Libro-de-Álgebra-Lineal.pdf")
Sucursal1= Sucursal("Sucursal Benito")
Sucursal2= Sucursal("Sucursal Juárez")
Bibliotecario1.agregar_material(Libro1, Sucursal1)
Catalogo1=Catalogo(Sucursal1)
Catalogo2= Catalogo(Sucursal2)
Catalogo1.agregar_material(Revista1)
Catalogo1.agregar_material(MaterialDigital1)
Usuario1.consultar_catálogo(Catalogo1)
Prestamo1= Prestamo(Usuario1, Libro1, "10/2/2025")
Usuario1.recibir_material(Libro1)
Usuario1.devolver_material(Libro1, "20/2/2025")
Catalogo1.buscar_material("chismes")
Catalogo1.buscar_material("álgebra")
Catalogo1.buscar_material("Lorax")
```

✓ 0.0s

```
... Se registró correctamente a Fernanda con correo de contacto fer123@buap.mx
Se registró correctamente a Carlos con correo de contacto carlos321@buap.mx
Catálogo de Sucursal Benito:
Título: Chismes |Edición: II |Periodicidad: Mensual |Estado: disponible
None
Título: Libro de Álgebra Lineal |Tipo de archivo: PDF |Enlace: Libro-de-Álgebra-Lineal.pdf
None
Estado de El lorax: prestado
Fernanda ha recibido el material: El lorax
Estado de El lorax: disponible
Fernanda ha devuelto el material: El lorax
Fernanda ha devuelto el material El lorax con 3 días de retraso
Fernanda tiene una penalización de $9.
Resultados de búsqueda para 'chismes':
Título: Chismes |Edición: II |Periodicidad: Mensual |Estado: disponible
- None
Resultados de búsqueda para 'álgebra':
Título: Libro de Álgebra Lineal |Tipo de archivo: PDF |Enlace: Libro-de-Álgebra-Lineal.pdf
- None
Resultados de búsqueda para 'Lorax':
No se encontraron materiales que coincidan con su búsqueda
```

- Liga Github:

<https://github.com/LuciaMaRu/Practical1U1>