

**Universidad
Autónoma
Metropolitana**



Casa abierta al tiempo **Azcapotzalco**

INGENIERÍA EN COMPUTACIÓN

PATRONES DE DISEÑO DE SOFTWARE

DRA. MARIA LIZBETH GALLARDO LÓPEZ

SISTEMA DE VENTA EN LÍNEA DE PIZZAS PERSONALIZADAS

ZURITA PRADO ISAI OBED

2223029036

24 DE NOVIEMBRE DEL 2025

Tabla de contenido

<i>Descripción del problema.....</i>	<i>3</i>
<i>Análisis del problema.....</i>	<i>3</i>
<i>Diagrama de casos de uso.....</i>	<i>4</i>
<i>Escenario de uso.....</i>	<i>4</i>
<i>Diagrama de dominio.....</i>	<i>6</i>
<i>Diagrama de clases</i>	<i>6</i>

Descripción del problema

Una pizzería busca modernizar su operación implementando un sistema de ventas de pizzas personalizadas, donde se requiere automatizar el flujo completo del pedido para mejorar la experiencia. El problema central radica en gestionar correctamente la complejidad de crear una pizza con múltiples combinaciones de ingredientes, calcular su costo y mantener informado al cliente en tiempo real sobre el estado de su orden. El sistema debe simular la lógica de preparación y mostrar gráficamente esta evolución.

Análisis del problema

Tabla de frases nominales

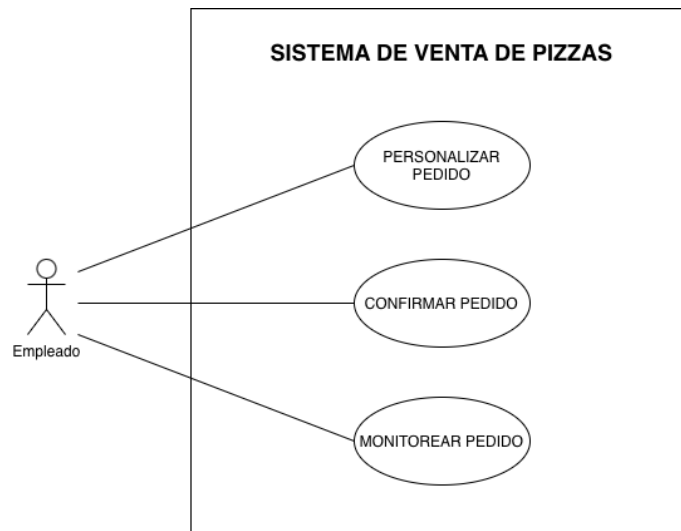
Frase nominal	Clase	Atributo	Pertenece a
Pizza personalizada	Pizza	x	Pizza
Tipo de masa	x	✓	Pizza
Tipo de salsa	x	✓	Pizza
Tipo de queso	x	✓	Pizza
Ingredientes adicionales	x	✓	Pizza
Condimentos	x	✓	Pizza
Tipo de orilla	x	✓	Pizza
Pedido	Pedido	x	Pedido
Costo final	x	✓	Pedido
Estado del pedido	x	✓	Pedido
Cliente	Cliente	x	Cliente
Nombre del cliente	x	✓	Cliente
Notificación	Notificación	x	Notificación
Interfaz de selección	PantallaSeleccion	x	PantallaSeleccion
Interfaz de seguimiento	PantallaSeguimiento	x	PantallaSeguimiento
Simulador de preparación	Cocina	x	Cocina

Tabla de frases verbales

Frase verbal	Método	Pertenece a
Seleccionar ingredientes	agregarIngrediente	Pizza
Seleccionar masa	seleccionarMasa	Pizza
Seleccionar salsa	seleccionarSalsa	Pizza
Calcular costo final	calcularCosto	Pedido
Registrar pedido	registrarPedido	Pedido

Preparar pizza / Simular preparación	prepararPizza	Cocina
Actualizar estado	actualizarEstado	Pedido
Notificar al cliente	notificarCliente	Notificación
Mostrar evolución	actualizarVista	PantallaSeguimiento
Tomar pedido	iniciarPedido	PantallaSeleccion

Diagrama de casos de uso



Escenario de uso

Título: Realizar pedido de pizza personalizada

Actor: Empleado (quien usa la interfaz gráfica para personalizar la pizza y confirmar la orden)

Descripción: El empleado personaliza una pizza paso a paso seleccionando diferentes ingredientes y modificadores, al confirmarla, el sistema calcula el costo y comienza la simulación de la preparación de la orden en cocina, actualizando la interfaz gráfica dependiendo del estado de la preparación.

Flujo normal:

1. El empleado inicia la aplicación y accede a la pantalla de nuevo pedido.
2. A través de la UI selecciona los diferentes ingredientes y modificadores.
3. El sistema muestra el costo total de la pizza personalizada.
4. El empleado confirma el pedido, el sistema registra la orden y el pedido recibe el status de recibido.

5. Después, el sistema actualiza automáticamente el paso del tiempo a través de los diferentes estados, como “en preparación, horneando, listo para entrega, entregado”
6. En cada cambio de estado la UI se actualiza y se genera la notificación al usuario(cliente).

Posibles patrones

Builder (creacional)

Se usaría para la construcción de la pizza, ya que el usuario debe seleccionar múltiples elementos paso a paso (masa, salsa, ingredientes) y así evitar un constructor extenso y confuso ya que el objeto Pizza podría tener una configuración combinatoria muy alta.

Factory Method (creacional)

Se implementaría para delegar la creación de los distintos tipos de ingredientes (carne, vegetales, condimentos), permitiendo que el sistema pueda instanciar las clases correctas sin acoplar el código a implementaciones específicas y permitiría posibles nuevos extras.

State (comportamiento)

Se usaría para gestionar el ciclo de vida y los diferentes tipos de estado que puede tener un pedido, permitiendo que el objeto cambie su comportamiento automáticamente, haciendo que no se necesite el uso de condicionales que puede hacer complejo el código y asegurando que la transición entre fases siga el ciclo de vida definido.

Observer (comportamiento)

Se utilizaría para cumplir con el requisito de la notificación y como ayudante de la UI, permitiendo que la interfaz “se suscriba” a los cambios de estado de Pedido y se actualice en tiempo real, manteniendo desacoplada la lógica del negocio de la UI.

Singleton (creacional)

Una sola instancia de Cocina, asegurando que esta única instancia se la encargada de gestionar la cola de pedidos, garantizando así sincronización adecuada de las órdenes.

Proxy (estructural) + Adapter (estructural)

Se implementaría la combinación de estos patrones para los pagos, donde el proxy actúa como un filtro de seguridad que valida los datos bancarios y posteriormente delega la transacción al adapter, la cual “traduce” el pago al formato del sistema bancario.

Diagrama de dominio

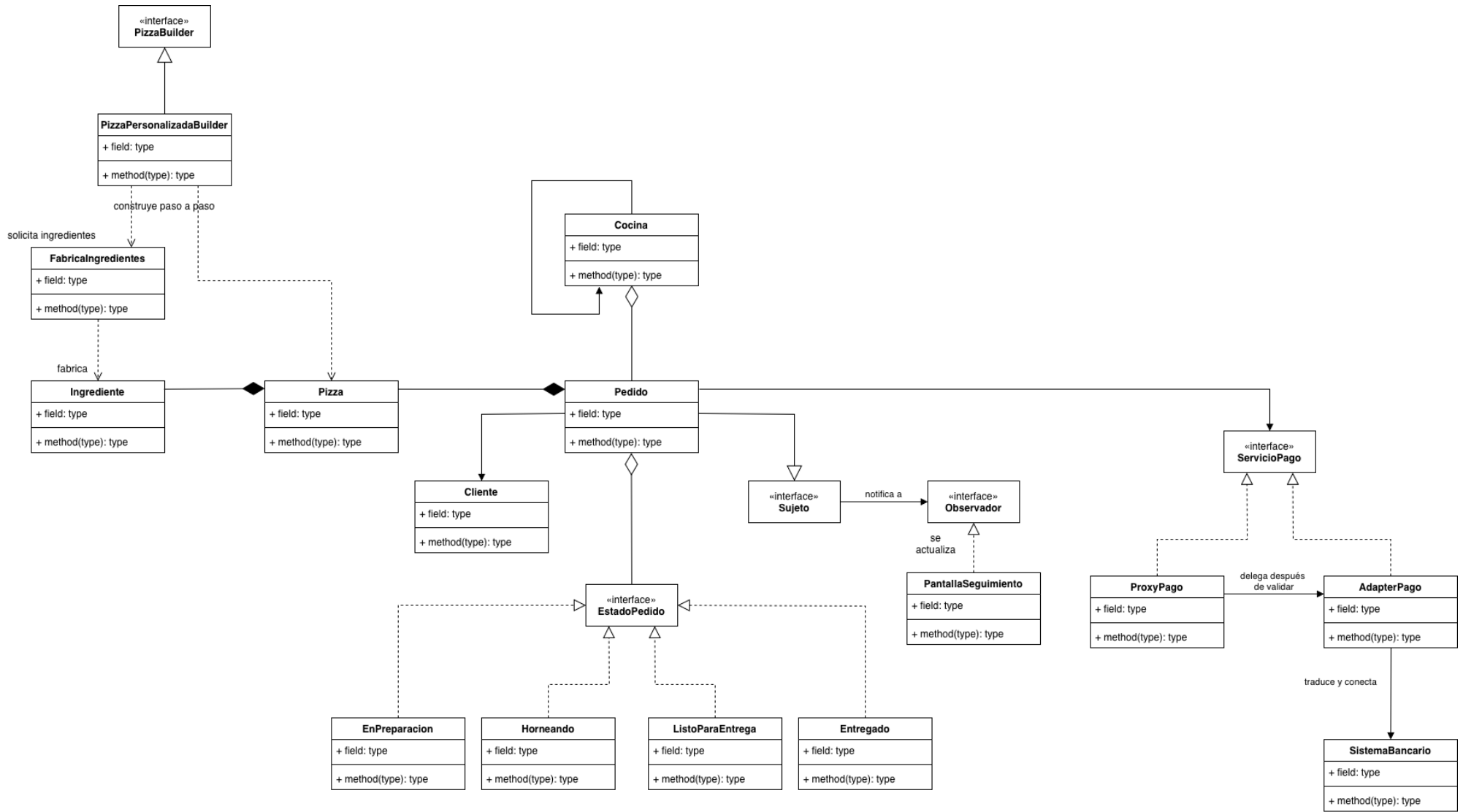
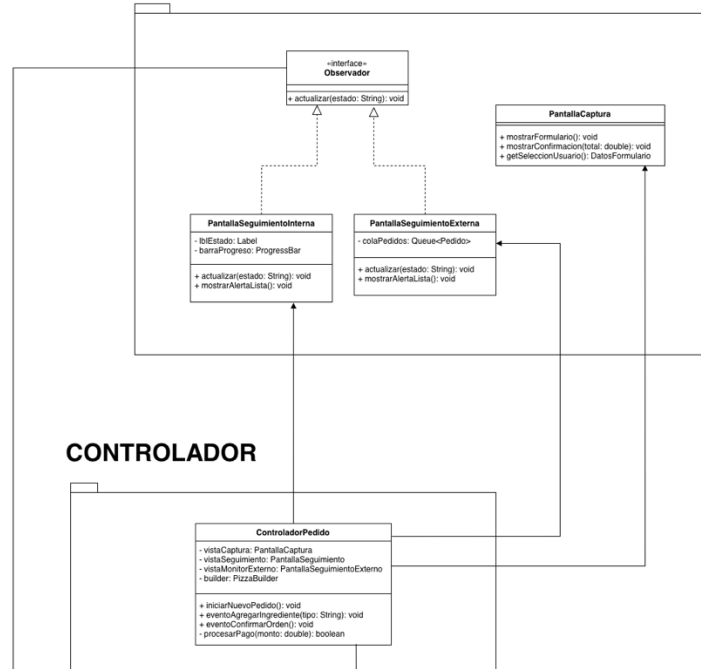
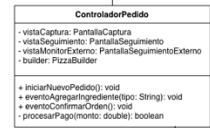


Diagrama de clases

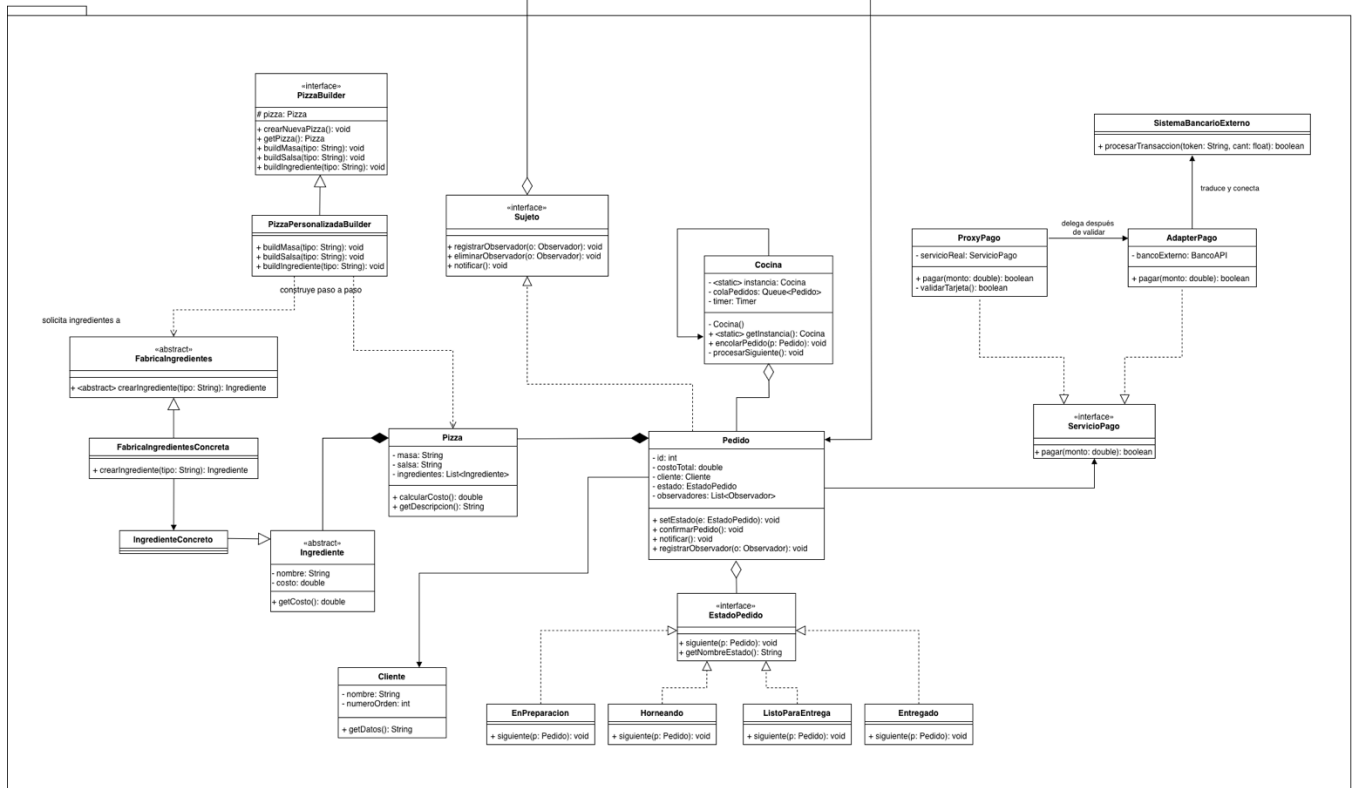
VISTA



CONTROLADOR



MODELO



Mockups de la UI

Pantalla para toma de órdenes

SISTEMA DE PIZZAS PERSONALIZADAS

Masa

Salsa

Ingredientes

Orilla

Extras

Selecciona los ingredientes adicionales

☒ Pepperoni

☒ Pimientos

☐ Aceitunas

☐ Tocino

RESUMEN DE ORDEN

Cliente: Juan Pérez

Número de orden: 008

Pizza grande

Masa tradicional

Salsa tomate

Queso mozzarella

Pepperoni

Pimientos

TOTAL: \$254.00

PROCEDER AL PAGO

Pantalla de cocina

SISTEMA DE PIZZAS PERSONALIZADAS - COLA DE PEDIDOS

ORDEN 007

Cliente: Fulano

Pizza mediana

Masa de sartén

Salsa tomate

Queso mozzarella

Pepperoni

Tocino

LISTA PARA ENTREGA

ORDEN 008

Cliente: Juan Pérez

Pizza grande

Masa tradicional

Salsa tomate

Queso mozzarella

Pepperoni

Pimientos

HORNEANDO

ORDEN 009

Cliente: Juan Pérez

Pizza grande

Masa Italiana

Salsa tomate

Orilla de queso moz:

Pepperoni

EN PREPARACIÓN

Pantalla externa, para que el cliente visualice los pedidos listos

