

Camilo Andres Castro Acosta

ACTIVIDAD 1 — Abstracción

1. Objetivo del Sistema

Facilitar la gestión digital de pedidos en una cafetería universitaria, permitiendo a estudiantes y personal ordenar, pagar y recibir notificaciones sobre sus pedidos de forma eficiente.

2. Actores

Actor	Descripción
Cliente	Estudiante o personal que realiza pedidos
Cajero/Admin	Gestiona menú y monitorea pedidos
Cocinero	Prepara los pedidos y actualiza estado
Sistema de Pagos	Procesa transacciones financieras
Sistema de Notificaciones	Envía alertas a clientes

3. Funciones Principales

*Visualizar menú digital

*Crear y personalizar pedidos

*Procesar pagos en línea

*Notificar estado de preparación

*Entregar pedido y confirmar recepción

4. Límites del Sistema

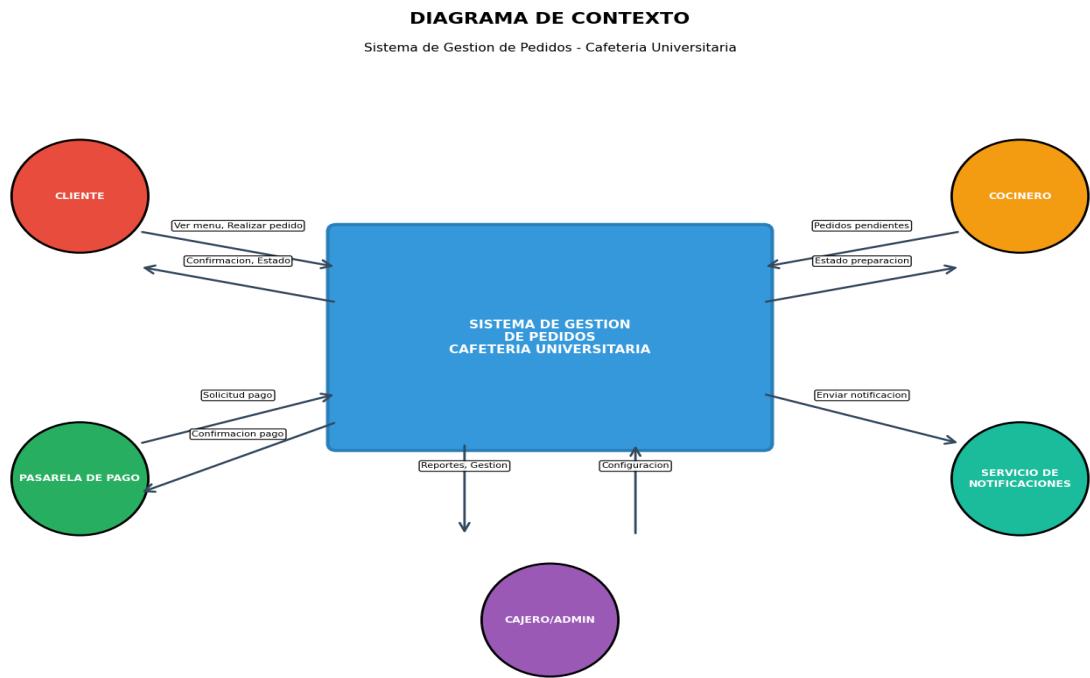
*No incluye inventario de proveedores

*No gestiona nómina del personal

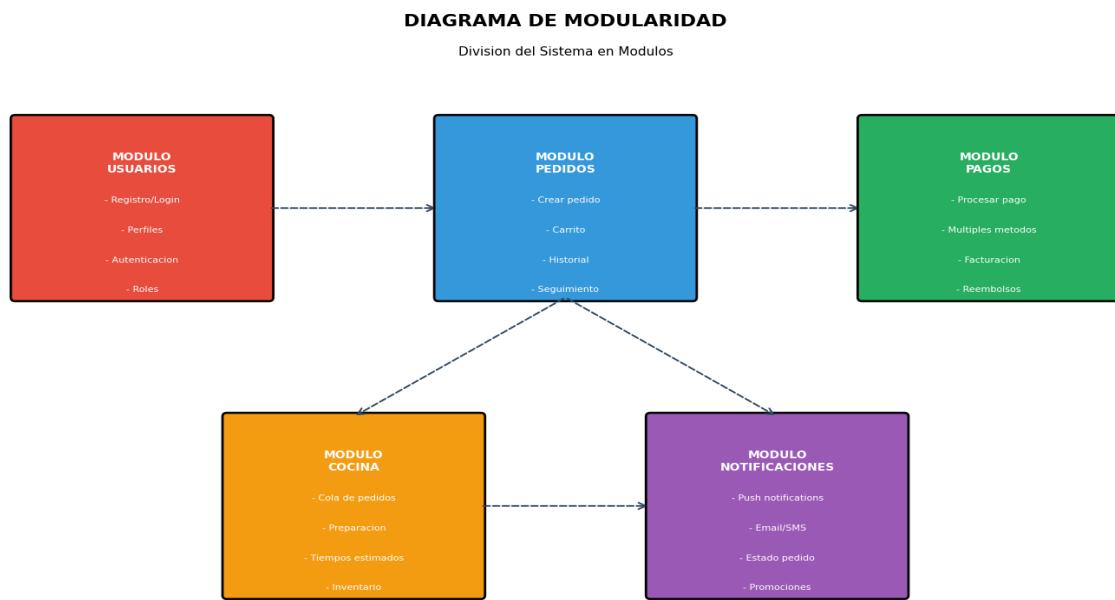
*No integra con sistemas académicos universitarios

*Solo operación dentro del campus

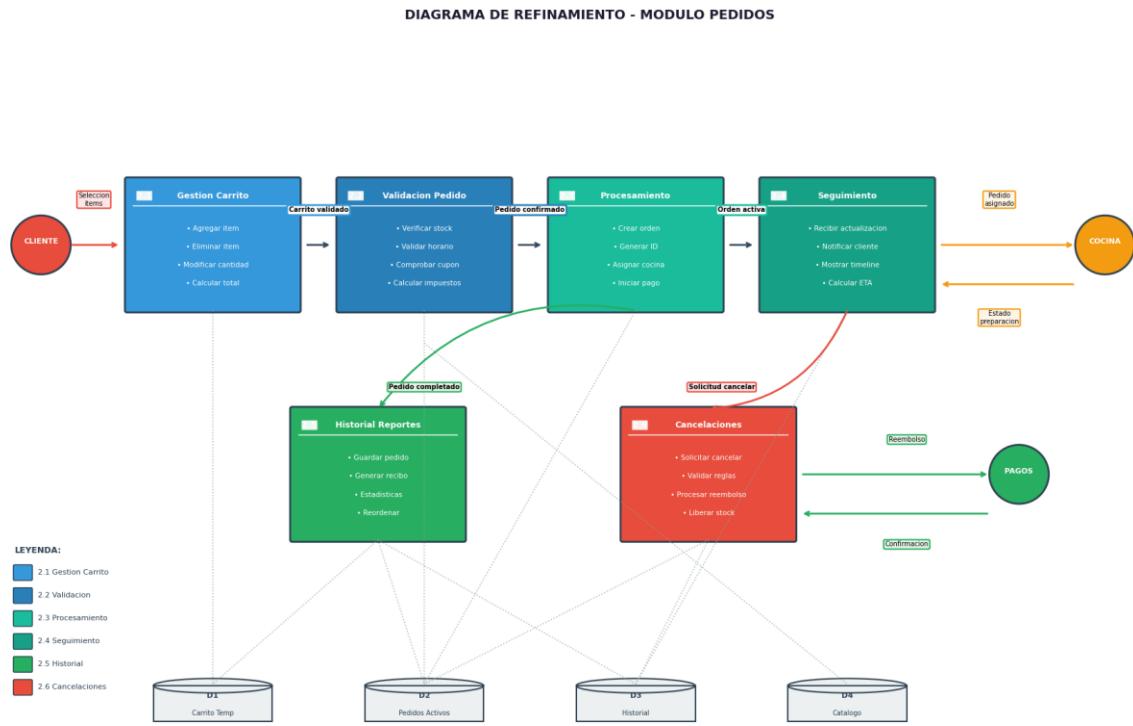
1. DIAGRAMA DE CONTEXTOS :



2. DIAGRAMA DE MODULARIDAD:



3. DIAGRAMA DE REFINAMIENTOS



4 . Aplicación de Principios

1. ¿Cómo el diseño es flexible?

Para que el sistema sea flexible lo que hicimos fue separar todo en partes bien definidas. Cada módulo se encarga de una sola cosa, así si necesitamos cambiar algo no se rompe todo el programa.

2. ¿Cómo permite pruebas?

Lo bueno de cómo quedó estructurado es que se puede probar todo sin tener que abrir la ventana gráfica. Como la lógica está separada de los botones y ventanas, podemos hacer tests unitarios directamente.

3. ¿Cómo reutiliza componentes?

El AlmacenesDatos es como una base de datos compartida que usan todos los módulos. No necesitamos crear una conexión a base de datos real, todo está en memoria y cualquier módulo puede leer o escribir ahí.

Las clases como Producto, ItemCarrito, Pedido y Factura son simples , entonces las podemos usar en cualquier otro proyecto sin modificarlas mucho. Si después hacemos una app móvil o una versión web, esas mismas clases sirven.

4. ¿Qué pasaría si crece a varias sedes?

Si la cafetería se expande a otros campus, habría que hacer algunos cambios:

Datos: Lo primero es que cada sede necesita su propia base de datos. En lugar de tener un solo AlmacenesDatos, tendríamos uno por sede (Norte, Sur, Este). Cada uno con su propio catálogo, precios y stock.

Configuración: Cada sede podría tener menús diferentes. Por ejemplo, la sede Norte podría tener más opciones de desayuno porque hay más clases en la mañana. Eso se manejaría con una clase de configuración por sede.

Sincronización: Para que la universidad vea reportes de todas las sedes, necesitamos una sede central (HQ) que reciba la información. Los pedidos completados se enviarían ahí para tener estadísticas consolidadas.

5. DIAGRAMA DE ARQUITECTURA:

