**Sistema Web Básico para Simulación de Restaurante**

Informe Técnico

Diego Alejandro Martinez Alvarado

Segundo Semestre

Desarrollo de Software

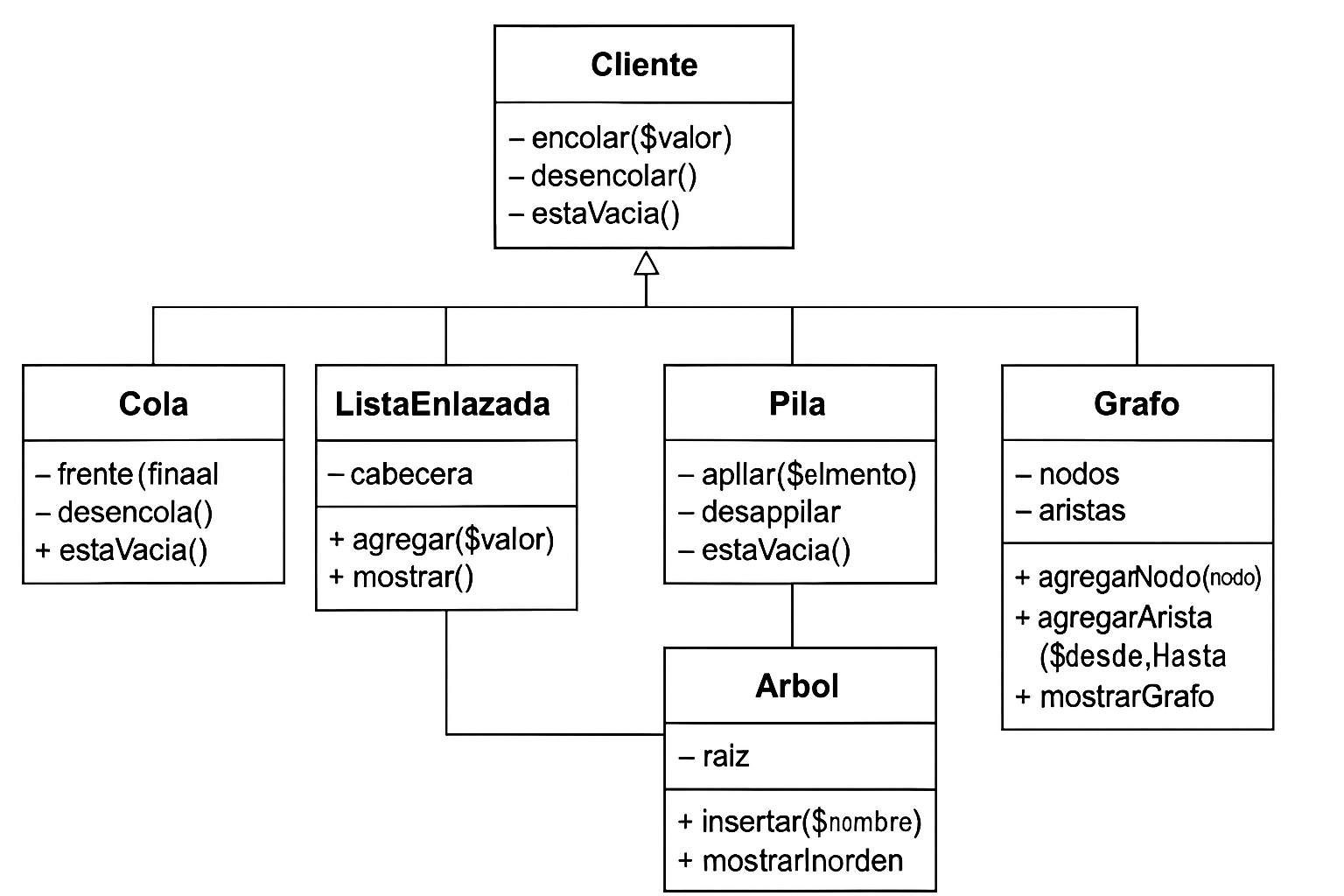
Fecha: 18 de Mayo de 2025

# Introducción

Este documento presenta el desarrollo de un sistema web básico implementado en PHP puro, que simula un restaurante donde múltiples clientes llegan, ordenan sus comidas y se van. El sistema integra diversas estructuras de datos fundamentales como arreglos, listas enlazadas, pilas, colas, árboles y grafos.

# 2. Diseño y Estructuras de Datos

Para modelar el comportamiento del restaurante, se implementaron las siguientes estructuras:  
- Cola: Para manejar la fila de clientes esperando ser atendidos.  
- Lista Enlazada: Para representar el pedido de cada cliente.  
- Pila: Para almacenar los platos servidos, simulando el orden de entrega.  
- Árbol: Para llevar un historial ordenado de clientes atendidos.  
- Grafo: Para modelar la distribución y conexiones entre las mesas del restaurante.  
- Arreglos: Para almacenar el menú de comidas disponible y la lista inicial de clientes.



### Clases Principales

#### 1. ****Cliente****

* Es la clase principal que **coordina el sistema**.
* Interactúa con todas las demás estructuras: Cola, ListaEnlazada, Pila, Árbol y Grafo.
* No almacena datos directamente, pero **invoca métodos** de las estructuras para simular la llegada, atención y salida del cliente.

### 📥 Cola (Queue)

* **Responsable de gestionar la fila** de clientes en orden de llegada (FIFO).
* Métodos:
  + encolar(valor) – Agrega un cliente.
  + desencolar() – Atiende al primero.
  + estaVacia() – Verifica si hay clientes.

### 🍽️ ListaEnlazada (Pedidos)

* Almacena **los pedidos de cada cliente**.
* Cada nodo representa un plato.
* Métodos:
  + agregar(pedido) – Añade un plato.
  + mostrar() – Lista los pedidos.

### 🔁 Pila (Platos Servidos)

* Registra el **orden en que se sirven los platos**, tipo LIFO.
* Métodos:
  + apilar(plato) – Añade plato servido.
  + desapilar() – Retira el último plato.
  + verPila() – Muestra platos servidos.

### 🌳 Árbol (Historial de Clientes)

* Guarda los **nombres de clientes atendidos** en un árbol binario.
* Permite orden alfabético o por prioridad.
* Métodos:
  + insertar(cliente)
  + inOrden (), preOrden(), postOrden() – para mostrar.

### 🪑 Grafo (Conexión de Mesas)

* Simula **mesas conectadas entre sí** (ej: por cercanía).
* Cada nodo es una mesa.
* Métodos:
  + agregarMesa(nombre)
  + conectarMesas(m1, m2)
  + verConexiones(mesa)

### 🔄 Relaciones entre clases

* **Cliente** es el núcleo que:
  + Encola a nuevos clientes (Cola).
  + Crea pedidos (ListaEnlazada).
  + Registra platos servidos (Pila).
  + Inserta historial (Árbol).
  + Muestra ubicación o conexiones de mesas (Grafo).

# 3. Implementación

El sistema fue desarrollado utilizando PHP puro, distribuyendo el código en módulos independientes para cada estructura de datos. Se emplearon arrays simples para mantener los datos en memoria durante la ejecución, evitando la complejidad de bases de datos o archivos externos.

Los archivos principales incluyen:  
- Cola.php  
- ListaEnlazada.php  
- Pila.php  
- Arbol.php  
- Grafo.php  
- cliente.php (archivo principal que integra todas las estructuras)  
  
Cada módulo contiene métodos para operar su estructura, y están comentados para facilitar la comprensión.

# 4. Ejecución y Resultados

A continuación, se presentan ejemplos de la salida generada por el sistema al ejecutar la simulación en consola:  
  
🧍 Cliente atendido: Ana  
🍽 Pedido:  
 - Hamburguesa  
 - Refresco  
 - Papas fritas  
✅ Pedido completo para Ana  
  
📦 Platos servidos (últimos primero):  
 - Luis: Pollo frito  
 - Ana: Papas fritas

# 5. Conclusión

El Proyecto permitió poner en práctica la implementación y uso de estructuras de datos fundamentales en PHP, simulando un escenario práctico y cotidiano. El sistema es funcional y puede ampliarse para incorporar manejo de sesiones, persistencia de datos o interfaz gráfica, aumentando su realismo y utilidad.

