

## DOCUMENTO DE DISEÑO

Restaurante La Casona: Innovación, creatividad y eficiencia

Versión: 0.1

---

**Team Beam**

Álvarez Bonilla Francisco Javier  
Hernández Hernández Jhoel Boset  
Nava Rivera Diego Alberto  
Palacios Medel Aldo  
Pérez Pavón Andy  
Viveros Reyes Haniel Antonio

**Documento de diseño**

Restaurante La Casona: Innovación, creatividad y eficiencia  
Puebla 21 de noviembre de 2023

# REVISIÓN Y CONTROL DE CAMBIOS

## Revisión y Versiones

Nombres y Apellidos	Cargo	Versión Aprobada	Fecha
Viveros Reyes Hacial Antonio	Director del Proyecto	0.1	23/11/23

## Control de Cambios

Fecha	Autor	Versión	Descripción del cambio
27/11/2023	Álvarez Bonilla Francisco Javier Hernández Hernández Jhoel Boset Nava Rivera Diego Alberto Palacios Medel Aldo Pérez Pavón Andy Viveros Reyes Hacial Antonio	0.1	

## TABLA DE CONTENIDO

<b>DOCUMENTO DE DISEÑO .....</b>	<b>1</b>
<b>REVISIÓN Y CONTROL DE CAMBIOS .....</b>	<b>3</b>
<b>Resumen: .....</b>	<b>5</b>
<b>1. Generalidades.....</b>	<b>5</b>
1.1.    Propósito del sistema.....	5
1.2.    Objetivos del diseño .....	6
1.3.    Definiciones, estereotipos UML y abreviaturas .....	7
1.4.    Referencias .....	9
<b>2. Arquitectura del sistema.....</b>	<b>9</b>
2.1.    Arquitectura propuesta.....	9
2.1.1.    Visión global. ....	9
2.1.2.    Diseño de la Arquitectura .....	10
2.1.3.    Diseño de los subsistemas.....	13
2.1.4.    Vista de casos de uso .....	14
2.1.4.1.    Vista lógica.....	15
<b>3. Diseño detallado.....</b>	<b>19</b>
3.1.    Introducción.....	19
3.2.    Diseño de casos de uso reales.....	20
3.3.    Diseño detallado de las clases. ....	24
<b>4. Diseño de la arquitectura de módulos del sistema.....</b>	<b>27</b>
<b>5. Diseño físico de datos .....</b>	<b>28</b>
<b>6. Modelo de datos y diccionario de datos.....</b>	<b>30</b>
6.1. Modelo Entidad – Relación (Diagrama de Clases) .....	33
.....	33
6.2. Diseño de datos .....	33
<b>7. Prototipo de pantallas.....</b>	<b>41</b>
1. Pantallas de Gestión de Mesas: .....	41
2. Pantallas de Peticiones y Cocina:.....	41
3. Pantallas de Facturación:.....	41
4. Pantallas de Aprovisionamiento:.....	41
5. Pantallas de Consumo de Alimentos: .....	41
6. Pantallas de Seguridad: .....	41
7. Pantallas de Comunicación:.....	41
8. Pantallas de Base de Datos:.....	41

## Resumen:

Este documento representa un plan detallado de diseño de software diseñado para implementar un sistema integral que gestionará eficientemente el servicio de restauración en el establecimiento gastronómico conocido como Restaurante La Casona. El objetivo principal es proporcionar una solución efectiva que optimice todas las operaciones del restaurante, desde la creación de menús hasta la facturación. Se abordan aspectos clave, como la recepción de pedidos, gestión de cocina, entrega de platos, aprovisionamiento y seguimiento del consumo de alimentos.

En las secciones subsiguientes, se ofrece una visión general del sistema o subsistema a desarrollar, junto con las decisiones y restricciones de diseño consideradas. También se incluye una lista completa de las definiciones de términos, acrónimos y abreviaturas utilizados en el documento, así como una lista de documentos de referencia. Por último, se proporciona un resumen que esquematiza la estructura y organización general de este documento de diseño para facilitar su comprensión y navegación.

## 1. Generalidades

### 1.1. Propósito del sistema.

El sistema tiene como propósito proporcionar una solución integral para la administración eficiente del servicio de comida en el restaurante, centrada en la creación de una experiencia óptima tanto para el personal como para los clientes. La arquitectura global del sistema se concibe con el objetivo de cumplir con requisitos funcionales y no funcionales específicos, garantizando eficiencia, usabilidad y seguridad.

#### Objetivos Generales del Diseño:

La arquitectura del sistema se diseñará con los siguientes objetivos:

- **Interfaz Amigable para Meseros en Tabletas:** Desarrollar una interfaz intuitiva y fácil de usar en tabletas para agilizar la toma de pedidos y la interacción con el sistema por parte de los meseros.
- **Mensajes de Error Explicativos:** Implementar mensajes de error claros y explicativos para facilitar una solución rápida y eficiente ante posibles problemas o malentendidos en la operación del sistema.
- **Gestión Eficiente de Pedidos en Horas Pico:** Garantizar la gestión eficiente de pedidos incluso durante periodos de alta demanda, optimizando el rendimiento del sistema para mantener la calidad del servicio.
- **Respuesta del Sistema en Menos de 3 Segundos:** Asegurar que la respuesta del sistema a consultas de menús y pedidos sea inferior a 3 segundos para proporcionar una experiencia fluida y sin demoras.
- **Protección de Configuración con Credenciales de Administrador:** Establecer medidas de seguridad para proteger la configuración del sistema, requiriendo credenciales de administrador • Lograr una mejora del 20% en la eficiencia de la gestión del restaurante: Se buscará una mejora sustancial en la eficiencia de la gestión del restaurante. Esto significa que, en comparación con los procesos manuales tradicionales, el nuevo sistema permitirá realizar tareas de gestión de manera más eficiente. Se medirá el rendimiento antes y después de la implementación para evaluar el logro de esta mejora.

- **Reducir los errores de pedidos y pagos en un 15%:** La implementación de la aplicación móvil para tomar pedidos deberá reducir la probabilidad de errores en la toma y transmisión de pedidos, así como en la facturación y el procesamiento de pagos. Se medirá la tasa de error antes y después de la implementación para evaluar el progreso.
- **Mantener un control de inventario de alimentos en tiempo real:** La aplicación web permitirá mantener un control de inventario de alimentos en tiempo real. Esto significa que, en cualquier momento, el personal podrá conocer el nivel de existencias de ingredientes y productos, lo que permitirá una mejor planificación y gestión de compras. La capacidad de mantener el inventario actualizado en tiempo real será un logro importante.
- **Permitir una variedad de métodos de pago para los clientes:** Se buscará ampliar las opciones de pago para los clientes. La aplicación móvil y la aplicación web deberán admitir al menos tres métodos de pago diferentes, como efectivo, tarjetas de crédito/débito y pago en línea, entre otros. Esto mejorará la comodidad de los clientes y la flexibilidad del restaurante.

## 1.2. Objetivos del diseño

La planificación del sistema se centra en satisfacer por completo los requisitos tanto funcionales como no funcionales, esenciales para asegurar una gestión eficiente del servicio en el restaurante. Los objetivos específicos que dirigirán el proceso de diseño son:

- **Eficiencia en la Interfaz para Meseros:**
  - **Objetivo:** Garantizar que la interfaz en tabletas para meseros sea eficiente y fácil de usar.
  - **Justificación:** Mejorar la eficiencia operativa y la experiencia del usuario para el personal del restaurante, optimizando la toma de pedidos y la interacción con el sistema.
- **Claridad en los Mensajes de Error:**
  - **Objetivo:** Implementar mensajes de error claros y explicativos.
  - **Justificación:** Facilitar la resolución rápida de problemas, reduciendo el tiempo de inactividad y mejorando la eficiencia operativa.
- **Gestión Efectiva en Horas Pico:**
  - **Objetivo:** Garantizar la gestión eficiente de pedidos en períodos de alta demanda.
  - **Justificación:** Mantener la calidad del servicio incluso en momentos de mayor actividad, asegurando una experiencia positiva para los clientes.
- **Respuesta Rápida del Sistema:**
  - **Objetivo:** Lograr una respuesta del sistema en menos de 3 segundos para consultas de menús y pedidos.
  - **Justificación:** Optimizar la experiencia del usuario, proporcionando respuestas rápidas y evitando posibles frustraciones.
- **Seguridad y Protección:**
  - **Objetivo:** Proteger la configuración del sistema con credenciales de administrador y almacenar datos del cliente de manera segura con cumplimiento de privacidad.
  - **Justificación:** Garantizar la integridad del sistema y la confidencialidad de la información del cliente, cumpliendo con estándares de seguridad y privacidad.
- **Disponibilidad del Sistema:**
  - **Objetivo:** Mantener una disponibilidad del sistema del 99% durante el horario comercial.
  - **Justificación:** Asegurar que el sistema esté disponible cuando sea necesario para evitar interrupciones en las operaciones del restaurante.
- **Copias de Seguridad Automáticas:**
  - **Objetivo:** Implementar copias de seguridad automáticas para una recuperación rápida.
  - **Justificación:** Minimizar el riesgo de pérdida de datos y garantizar la continuidad del servicio mediante la rápida recuperación de información en caso de incidentes.

## **Influencia en el Logro de Objetivos (Requisitos Funcionales y No Funcionales):**

- **Alineación Operativa:**
  - **Garantía de Funcionalidad:** Asegurar que el diseño satisfaga los requisitos operativos, desde la toma de pedidos hasta el control de inventarios.
- **Cumplimiento de Experiencia del Usuario:**
  - **Presentación y Gestión:** Asegurar la efectiva implementación de requisitos no funcionales, abarcando la presentación de la carta digital, actualización de información sobre platillos y manejo de sustitución de ingredientes.

## **Decisiones Estratégicas y Enfoques:**

- **Enfoque en la Protección Integral:**
  - **Seguridad General:** Colocar un marcado énfasis en la seguridad global del sistema, priorizando la protección de datos y la privacidad del usuario.
- **Transición Fluida y Sin Interrupciones:**
  - **Implementación Continua:** Planificar la introducción del software de manera que evite interrupciones, con procedimientos establecidos para asegurar una transición sin contratiempos y garantizar una implementación continua.
- **Optimización de Comunicación Interna:**
  - **Prioridad en la Interconexión de Funciones:** Resaltar la importancia de una comunicación efectiva entre el personal de recepción, cocina, servicio y gestión, enfocándonos en optimizar la interconexión de funciones.

## **1.3. Definiciones, estereotipos UML y abreviaturas**

### **Definiciones:**

- **Aplicación Móvil para Pedidos:** Herramienta dedicada que facilita a los clientes realizar pedidos de manera rápida y conveniente a través de dispositivos móviles.
- **Carta Digital Personalizada:** Interfaz electrónica que presenta el menú del restaurante La Casona, detallando cada platillo con descripciones y tiempos de preparación.
- **Confidencialidad:** Protección de la información sensible y privada del restaurante y sus clientes.
- **Eficiencia Operativa:** Mejora en la productividad y eficacia de las operaciones diarias del restaurante.
- **Entorno Presencial:** Configuración física del restaurante que habilita a meseros y clientes para realizar acciones similares a las de la aplicación, como la toma de órdenes en la mesa y la gestión de pedidos en tiempo real.
- **Estructura de Costos:** Desglose de los gastos asociados con la operación del restaurante, incluyendo costos de personal, alimentos e ingredientes, alquiler y gastos generales.
- **Fuentes de Ingresos:** Principales vías a través de las cuales el restaurante genera ganancias, como las ventas de alimentos y servicios de catering.
- **Gestión de Inventario y Proveedores:** Sistema de software que ofrece un control minucioso del inventario y facilita la gestión de relaciones con diferentes proveedores de ingredientes.
- **Gestión de Inventario:** Proceso de compra, registro y control de ingredientes para garantizar la disponibilidad en la preparación de platillos.
- **Ingeniería de Software:** Aplicación de principios de ingeniería al diseño, desarrollo, implementación y mantenimiento de software.
- **Innovación Tecnológica:** Implementación de soluciones tecnológicas para mejorar la eficiencia y la experiencia del cliente.
- **Interesados:** Personas o entidades que tienen un interés directo en el éxito del proyecto.
- **Marketing y Promoción:** Estrategias para aumentar la visibilidad del restaurante, incluyendo

publicidad y participación en eventos.

- **Menú y Precios:** Oferta de platillos y rangos de precios proporcionados a los clientes.
- **Mobile-Front-End (MFE):** Interfaz de usuario de una aplicación móvil diseñada para dispositivos móviles.
- **Modelo de Negocio:** Estructura que describe cómo opera y genera ingresos el restaurante La Casona.
- **Opciones de Pago:** Métodos de pago proporcionados a los clientes, incluyendo transferencias bancarias, tarjetas de crédito, débito y efectivo.
- **Operaciones Diarias:** Actividades rutinarias relacionadas con la atención al cliente, toma de pedidos y mantenimiento del restaurante.
- **Personal:** Conjunto de empleados, incluyendo chefs y personal de servicio, que contribuyen al funcionamiento del restaurante.
- **Pizarra Interactiva:** Herramienta integrada que optimiza la gestión operativa en la cocina, mostrando visualmente las órdenes en preparación y facilitando la asignación inteligente de recursos.
- **Plataforma Web para la Gestión Interna:** Ambiente en línea que simplifica la gestión interna del restaurante, permitiendo la actualización en tiempo real del menú, la programación de turnos y el control de inventario.
- **Protocolos:** Procedimientos y reglas establecidas para garantizar una transición suave hacia el nuevo sistema.
- **Rentabilidad:** Capacidad del restaurante para generar ganancias y beneficios económicos.
- **Retorno de la Inversión:** Beneficios económicos obtenidos como resultado de la inversión realizada en el proyecto.
- **Satisfacción del Cliente:** Nivel de satisfacción experimentado por los clientes al interactuar con el restaurante.
- **Segmento de Clientes:** Grupos específicos de personas a los que el restaurante dirige sus servicios y productos.
- **Seguridad Alimentaria:** Cumplimiento de normas y regulaciones para garantizar la seguridad y calidad de los alimentos servidos.
- **Sistema en Pleno Funcionamiento:** Estado en el que todos los componentes del software y hardware operan sin problemas.
- **Solutions Crafters:** Consultoría de software encargada de proporcionar la solución integral para la modernización del restaurante La Casona.
- **Tendencias del Mercado:** Cambios y preferencias actuales en la industria de la restauración que el restaurante debe tener en cuenta.
- **Ubicación Estratégica:** Posicionamiento del restaurante en una ubicación conveniente para atraer a turistas y residentes locales.
- **Ventaja Competitiva:** Característica o conjunto de características que otorgan al restaurante una posición favorable en comparación con la competencia.
- **Web-Back-End (WBE):** Parte de la aplicación web que maneja la lógica detrás de escena y funcionalidades internas.
- **Web-Front-End (WFE):** Interfaz de usuario de una aplicación web para la interacción directa con los usuarios.

## Acrónimos y Abreviaturas:

- **API:** Interfaz de Programación de Aplicaciones
- **CSS:** Hojas de Estilo en Cascada
- **DBMS:** Sistema de Gestión de Bases de Datos
- **ERP:** Planificación de Recursos Empresariales



- **HTTPS:** Protocolo de Transferencia de Hipertexto Seguro
- **PDV:** Punto de Venta
- **POS:** Punto de Venta
- **RF:** Requisito funcional
- **RNF:** Requisito no funcional
- **SQL:** Lenguaje de Consulta Estructurado
- **TI:** Tecnologías de la Información
- **UX:** Experiencia del Usuario

## 1.4. Referencias

Team Bean. "[Acta de constitución](#)". Proyecto: Restaurante La Casona: Innovación, creatividad y eficiencia, 2023

Team Bean. "[Documento de requerimientos](#)". Proyecto: Restaurante La Casona: Innovación, creatividad y eficiencia, 2023

## 2. Arquitectura del sistema.

### 2.1. Arquitectura propuesta

#### Estilo Arquitectónico: Arquitectura en Capas

La arquitectura propuesta para el sistema de gestión de servicios de restauración sigue un estilo arquitectónico en capas. Este enfoque proporciona una clara separación de responsabilidades, facilita la modularidad y permite un desarrollo más escalonado y mantenible.

<< Diagrama de actividades en la sección de "Tabla de Imágenes" >>

#### 2.1.1. Visión global.

#### Patrón Arquitectónico: Arquitectura en Capas

La estructura global de la arquitectura del sistema se compone de tres capas principales:

- **Capa de Presentación:**
  - *Descripción:* La Capa de Presentación es la interfaz directa con los usuarios finales, en este caso, los meseros que utilizan Tablet PCs. Es responsable de mostrar menús, registrar pedidos y gestionar la facturación. Actúa como el punto de entrada para la interacción del usuario.
- **Capa de Aplicación o Lógica de Negocio:**
  - *Descripción:* La Capa de Aplicación es el corazón del sistema, donde reside la lógica

empresarial. Contiene subsistemas que abordan funciones específicas como gestión de mesas, peticiones, cocina, entrega de platos, facturación, aprovisionamiento y consumo de alimentos. Coordina las operaciones y garantiza la coherencia de los datos.

- **Capa de Persistencia:**

- *Descripción:* La Capa de Persistencia se encarga del almacenamiento y recuperación de datos. Utiliza una base de datos relacional para almacenar información esencial, como detalles de mesas, pedidos, proveedores, etc. La comunicación con la Capa de Aplicación garantiza el acceso eficiente a los datos.

**Asignación de Funcionalidad:**

- *Capa de Presentación:*

- Muestra menús y recibe pedidos.
- Gestiona la interfaz de facturación.
- Comunica solicitudes a la Capa de Aplicación.

- *Capa de Aplicación o Lógica de Negocio:*

- Coordina la lógica principal del sistema.
- Subsistemas específicos para cada función del restaurante.
- Comunica con la Capa de Presentación y la Capa de Persistencia.

- *Capa de Persistencia:*

- Almacena y recupera datos críticos.
- Utiliza una base de datos relacional para la persistencia.
- Interactúa con la Capa de Aplicación para operaciones de lectura/escritura.

**Descripción de la Arquitectura Software:** La elección de una arquitectura en capas proporciona una clara separación de responsabilidades. La Capa de Presentación maneja la interacción con el usuario, la Capa de Aplicación se encarga de la lógica de negocio, mientras que la Capa de Persistencia administra el almacenamiento de datos. Este enfoque facilita el mantenimiento, la escalabilidad y la implementación de cambios sin afectar otras partes del sistema. Además, el patrón arquitectónico de Cliente-Servidor en Capas permite una distribución eficiente de la carga de trabajo y una mayor flexibilidad en el desarrollo y la evolución del sistema.

## 2.1.2. Diseño de la Arquitectura

**Descomposición en Subsistemas y Responsabilidades:**

- **Subsistema de Gestión de Mesas:**

- *Responsabilidades:*
  - Asignación de meseros a mesas.
  - Registro y consulta de ventas por mesa.
  - Coordinación con la Capa de Aplicación para actualizar datos.

- **Subsistema de Peticiones y Cocina:**

- *Responsabilidades:*
  - Registro y visualización de peticiones.
  - Coordinación de la elaboración de platos en la cocina.
  - Comunicación con la Capa de Aplicación para la gestión de pedidos.

- **Subsistema de Facturación:**
  - *Responsabilidades:*
    - Generación y gestión de facturas.
    - Coordinación con la Capa de Aplicación para obtener información de pedidos.
- **Subsistema de Aprovisionamiento:**
  - *Responsabilidades:*
    - Gestión de pedidos a proveedores.
    - Recepción de mercancía y facturas.
    - Coordinación con la Capa de Aplicación para actualizar inventarios.
- **Subsistema de Consumo de Alimentos:**
  - *Responsabilidades:*
    - Registro del consumo de alimentos.
    - Cálculo de unidades a descontar al final del día.
    - Coordinación con la Capa de Aplicación para mantener registros precisos.
- **Subsistema de Seguridad:**
  - *Responsabilidades:*
    - Implementación de mecanismos de autenticación.
    - Encriptación de datos sensibles.
    - Control de acceso basado en roles.
- **Subsistema de Comunicación:**
  - *Responsabilidades:*
    - Gestión de la comunicación entre la Capa de Presentación y la Capa de Aplicación.
    - Manejo de solicitudes y respuestas entre dispositivos Tablet PC y el servidor central.
- **Subsistema de Base de Datos:**
  - *Responsabilidades:*
    - Almacenamiento y recuperación eficiente de datos.
    - Mantenimiento de la consistencia de la base de datos.
    - Implementación de transacciones para garantizar la integridad de los datos.

Cada subsistema se encarga de funciones específicas del sistema, y la coordinación entre ellos se realiza a través de la Capa de Aplicación. Esta descomposición facilita la asignación de tareas y la modularidad del sistema, permitiendo un desarrollo y mantenimiento más eficientes.

### **Topología del Sistema (Diagrama de Despliegue):**

La topología del sistema describe la asignación del software al hardware y la interconexión entre los componentes. En este contexto, se presenta un diagrama de despliegue que representa cómo los distintos elementos del sistema se distribuyen físicamente.

### **Diagrama de Componentes:**

El diagrama de componentes muestra los principales componentes del sistema y sus interacciones. Cada capa de la arquitectura en capas se representa como un conjunto de componentes, indicando

la relación y dependencia entre ellos.

### **Gestión de la Persistencia:**

#### *Infraestructura de Almacenamiento:*

- Se utilizará una base de datos relacional (por ejemplo, MySQL, PostgreSQL) para la Capa de Persistencia.
- La base de datos contendrá tablas para mesas, pedidos, proveedores, etc.

#### *Datos Almacenados:*

- Información sobre meseros (DNI, apellidos, nombre).
- Detalles de mesas y asignación de meseros.
- Registros de pedidos, platos, y facturas.
- Datos de proveedores y alimentos.

### **Aspectos Globales y de Seguridad:**

#### *Políticas de Seguridad y Acceso:*

- Autenticación de meseros al acceder al sistema.
- Encriptación de datos sensibles, como información de clientes.
- Políticas de acceso basadas en roles (gerente, mesero, jefe de cocina).

#### *Sincronización y Concurrencia:*

- Mecanismos para garantizar la consistencia de datos en entornos concurrentes.
- Gestión de transacciones para operaciones críticas.

#### *Inicialización de Subsistemas:*

- Procedimientos de inicialización para asegurar un arranque seguro y eficiente del sistema.

### **Aspectos de Rendimiento y Tamaño:**

#### *Acuerdos de Nivel de Servicio (SLA):*

- Tiempos de respuesta aceptables para operaciones críticas como toma de pedidos, entrega de platos y facturación.
- Productividad definida en términos de operaciones por segundo o por período.

#### *Capacidad de Almacenamiento:*

- Se garantizará la capacidad de almacenamiento suficiente para la base de datos, considerando el crecimiento esperado de datos.

#### *Requisitos de Rendimiento:*

- El sistema debe cumplir con los tiempos de respuesta especificados en los SLA.
- Se establecerán métricas de rendimiento y se realizarán pruebas para validar el rendimiento del sistema bajo carga.

### 2.1.3. Diseño de los subsistemas

#### 1. Subsistema de Gestión de Mesas:

- **Servicios:**
  - Asignación de meseros a mesas.
  - Registro y consulta de ventas por mesa.
- **Interfaces:**
  - **Interfaz de Asignación de Mesas:**
    - Método: **asignarMeseroAMesa(meseroID, mesaNumero)**
  - **Interfaz de Registro de Ventas:**
    - Método: **registrarVenta(mesaNumero, detalleVenta)**

#### 2. Subsistema de Peticiones y Cocina:

- **Servicios:**
  - Registro y visualización de peticiones.
  - Coordinación de la elaboración de platos en la cocina.
- **Interfaces:**
  - **Interfaz de Registro de Peticiones:**
    - Método: **registrarPetition(mesaNumero, detallesPlatos)**
  - **Interfaz de Visualización de Peticiones:**
    - Método: **obtenerPeticionesPorMesa(mesaNumero)**
  - **Interfaz de Aviso de Finalización:**
    - Método: **avisarFinalizacionPlato(platoID)**

#### 3. Subsistema de Facturación:

- **Servicios:**
  - Generación y gestión de facturas.
- **Interfaces:**
  - **Interfaz de Generación de Facturas:**
    - Método: **generarFactura(mesaNumero, detallesVenta)**
  - **Interfaz de Gestión de Facturas:**
    - Método: **consultarFactura(facturalID)**

#### 4. Subsistema de Aprovisionamiento:

- **Servicios:**
  - Gestión de pedidos a proveedores.
  - Recepción de mercancía y facturas.
- **Interfaces:**
  - **Interfaz de Gestión de Pedidos:**
    - Método: **realizarPedido(proveedorID, detallesPedido)**
  - **Interfaz de Recepción de Mercancía:**
    - Método: **recibirMercancia(proveedorID, detallesMercancia)**
  - **Interfaz de Gestión de Facturas de Proveedores:**
    - Método: **gestionarFacturaProveedor(proveedorID, facturalID)**

#### 5. Subsistema de Consumo de Alimentos:

- **Servicios:**
  - Registro del consumo de alimentos.

- Cálculo de unidades a descontar al final del día.
- **Interfaces:**
  - **Interfaz de Registro de Consumo:**
    - Método: **registrarConsumoAlimentos(detalleConsumo)**
  - **Interfaz de Cálculo de Consumo Diario:**
    - Método: **calcularConsumoDiario()**

#### 6. Subsistema de Seguridad:

- **Servicios:**
  - Implementación de mecanismos de autenticación.
  - Encriptación de datos sensibles.
  - Control de acceso basado en roles.
- **Interfaces:**
  - **Interfaz de Autenticación:**
    - Método: **autenticar(usuario, contraseña)**
  - **Interfaz de Control de Acceso:**
    - Método: **verificarAcceso(rolUsuario, recurso)**

#### 7. Subsistema de Comunicación:

- **Servicios:**
  - Gestión de la comunicación entre la Capa de Presentación y la Capa de Aplicación.
- **Interfaces:**
  - **Interfaz de Envío de Solicitudes:**
    - Método: **enviarSolicitud(datosSolicitud)**
  - **Interfaz de Recepción de Respuestas:**
    - Método: **recibirRespuesta(datosRespuesta)**

#### 8. Subsistema de Base de Datos:

- **Servicios:**
  - Almacenamiento y recuperación eficiente de datos.
  - Mantenimiento de la consistencia de la base de datos.
  - Implementación de transacciones para garantizar la integridad de los datos.
- **Interfaces:**
  - **Interfaz de Acceso a Datos:**
    - Métodos para realizar operaciones de lectura y escritura en la base de datos.
    - Ejemplo: **realizarConsulta(SQL)**.

Estos subsistemas proporcionan servicios específicos y se comunican entre sí para realizar las funciones esenciales del sistema de gestión de servicios de restauración. Las interfaces definidas establecen cómo cada subsistema interactúa con los demás, permitiendo una integración efectiva de los componentes del sistema.

#### 2.1.4. Vista de casos de uso

Acrónimo RF	Nombre	Descripción
RF1	Iniciar Sesión	Todos los actores del sistema ingresan a la aplicación mediante un nombre de usuario y contraseña.
RF2	Abrir Orden	Los actores toman las ordenes de la mesa, las consultan y modifican.

RF3	Activar Mesa	Activa una mesa para indicar que hay clientes usándola.
RF4	Consultar Orden	Permite al actor consultar la orden requerida.
RF5	Cerrar Orden	El mesero cierra la orden y se emite la facturación.
RF6	Administrar Menú	Permite crear, modificar, eliminar y consultar los menús.
RF7	Despachar Orden	Permite despachar y consultar las órdenes emitidas por los meseros.
RF8	Consultar Reportes	Permite realizar consultas para obtener el resumen en forma de lista de las ventas y meseros de cada mesa.
RF9	Gestión de Mesas	El gerente configura el número de mesas y la asignación de mesero.
RF10	Gestión de Inventario	Permite llevar un control de los alimentos e insumos.
RF11	Cerrar Sesión	Permite a los actores del sistema cerrar la sesión actual.

#### 2.1.4.1. Vista lógica

Esta vista detalla las partes arquitectónicamente significativas del modelo de diseño, incluida la descomposición de los subsistemas en paquetes y la descripción de las clases relevantes con sus responsabilidades, relaciones, atributos y operaciones.

##### **Paquete: Gestión de Mesas (gestionMesas):**

*Descripción:* Este paquete se encarga de la asignación de meseros a mesas, la gestión de ventas por mesa y la coordinación de la información relacionada con las mesas.

*Clases:*

- **Mesa:**
  - *Responsabilidades:*
    - Almacenar información sobre una mesa específica.
  - *Atributos:*
    - NumeroMesa, MeseroAsignado, Factura.
  - *Operaciones:*

- AsignarMesero(MeseroID), RegistrarVenta(DetalleVenta).

- **Mesero:**

- *Responsabilidades:*
  - Almacenar información sobre un mesero.
- *Atributos:*
  - DNI, Apellidos, Nombre.
- *Operaciones:*
  - ConsultarVentas(MesaNumero).

**Paquete: Peticiones y Cocina (peticionesCocina):**

*Descripción:* Este paquete se ocupa de la gestión de peticiones, la visualización en la cocina y la elaboración de platos.

*Clases:*

- **Peticion:**

- *Responsabilidades:*
  - Almacenar detalles de una petición de un comensal.
- *Atributos:*
  - MesaNumero, DetallesPlatos.
- *Operaciones:*
  - EnviarPeticon(), VisualizarEnCocina(), AvisarFinalizacionPlato(PlatoID).

- **Plato:**

- *Responsabilidades:*
  - Representar un plato específico.
- *Atributos:*
  - Nombre, Tipo, TiempoElaboracion.
- *Operaciones:*
  - Elaborar(), ListoParaServir().

**Paquete: Facturación (facturacion):**

*Descripción:* Este paquete se encarga de la generación y gestión de facturas.

*Clases:*

- **Factura:**

- ***Responsabilidades:***
  - Almacenar detalles de una factura.
- ***Atributos:***
  - ID, MesaNumero, DetallesVenta.
- ***Operaciones:***
  - GenerarFactura(), ConsultarFactura().

**Paquete: Aprovisionamiento (aprovisionamiento):**

*Descripción:* Este paquete gestiona el aprovisionamiento de alimentos, la realización de pedidos a proveedores y la recepción de mercancía.



*Clases:*

- **PedidoProveedor:**
  - **Responsabilidades:**
    - Representa un pedido específico a un proveedor.
  - **Atributos:**
    - ProveedorID, DetallesPedido.
  - **Operaciones:**
    - RealizarPedido().
- **Mercancia:**
  - **Responsabilidades:**
    - Representa la mercancía recibida.
  - **Atributos:**
    - ProveedorID, DetallesMercancia.
  - **Operaciones:**
    - RecibirMercancia().

**Paquete: Consumo de Alimentos (consumoAlimentos):**

**Descripción:** Este paquete registra el consumo de alimentos y realiza cálculos al final del día.

*Clases:*

- **RegistroConsumo:**
  - **Responsabilidades:**
    - Almacenar detalles del consumo de alimentos.
  - **Atributos:**
    - DetalleConsumo.
  - **Operaciones:**
    - RegistrarConsumo().

**Paquete: Seguridad (seguridad):**

**Descripción:** Este paquete se ocupa de la implementación de mecanismos de autenticación, encriptación y control de acceso.

*Clases:*

- **Autenticacion:**
  - **Responsabilidades:**
    - Validar credenciales de usuarios.
  - **Atributos:**
    - UsuariosRegistrados.
  - **Operaciones:**
    - Autenticar().
- **ControlAcceso:**
  - **Responsabilidades:**
    - Gestionar políticas de control de acceso.
  - **Atributos:**
    - Roles, Recursos.

- *Operaciones:*
  - VerificarAcceso().

Estas clases y paquetes representan las partes arquitectónicamente significativas del sistema,

## 3. Diseño detallado

### 3.1. Introducción.

En esta sección, se presentarán las decisiones de diseño específicas tomadas para el sistema de gestión de restaurantes. Estas decisiones abarcan el enfoque de desarrollo elegido, consideraciones de rendimiento, estándares a seguir en el proceso de desarrollo y pautas para la creación de interfaces y codificación. El propósito de estas opciones es proporcionar una guía clara y detallada a los responsables del diseño e implementación del sistema, sirviendo como un plan claro para la construcción eficiente y coherente del proyecto.

#### 1. Modelo de Desarrollo:

Para facilitar la adaptabilidad continua a los cambios y asegura una mayor participación del cliente durante el proceso de desarrollo se optará por un modelo de desarrollo iterativo e incremental, permitiendo entregas planificadas para revisión y retroalimentación. El enfoque se centrará en la construcción gradual del sistema, priorizando la flexibilidad y la retroalimentación constante.

#### 2. Enfoque en el rendimiento:

Para asegurar las operaciones de manera eficiente en las áreas críticas del sistema, como la toma de pedidos y la facturación. Se llevará a cabo una optimización continua, considerando cuidadosamente el equilibrio entre espacio en memoria y tiempo de respuesta. La prioridad será el rendimiento en tiempo de respuesta para garantizar una experiencia de usuario eficiente.

#### 3. Estándares de interfaces:

- **Usuario:** Se establecerán estándares claros para las interfaces de usuario, asegurando una experiencia homogénea y amigable en todas las plataformas, especialmente en las tabletas utilizadas por el personal del restaurante.
- **Comunicación:** Utilizaremos protocolos de comunicación para que los diferentes partes del sistema se entiendan entre sí y se integren fácilmente con sistemas externos. Al seguir estándares reconocidos en la industria, aseguramos que el sistema sea flexible y sostenible a lo largo del tiempo.

#### 4. Modelo de desarrollo:

- **Convención de Nomenclatura:** Se adoptará una estrategia de nomenclatura precisa y comprensible para variables, funciones y clases. Esta elección no sólo optimiza la comprensión del código, sino que también promueve una colaboración fluida entre los desarrolladores para establecer una estructura de nombres clara y significativa.
- **Gestión de Situaciones Inesperadas:** Se instituirá un enfoque diligente para manejar situaciones imprevistas mediante la implementación de una estrategia eficaz de gestión de excepciones. Se establecerán procedimientos definidos y claros para la identificación, registro y gestión de errores durante la ejecución del sistema, fortaleciendo así su robustez y confiabilidad.

#### 5. Estándares de Codificación:

- **Coherencia en la escritura del código:** Se definirá y aplicará una guía coherente para escribir código a lo largo del proyecto, abordando elementos clave como la sangría y el manejo de espacios en blanco. Esta coherencia no sólo mejorará la estética del código, sino que también contribuirá significativamente a su legibilidad.
- **Documentación de código rigurosa:** Cada componente y módulo crítico del sistema se documentará minuciosamente. Esto significa incluir descripciones detalladas de las funciones, comentarios relevantes y explicaciones completas. Este enfoque tiene como objetivo hacer que el código sea más fácil de entender y mantener, proporcionando una referencia completa para quienes trabajan en el proyecto.

### 3.2. Diseño de casos de uso reales

<b>RF- 01</b>	Recepción de peticiones en las mesas	
<b>Actor Principal</b>	Mesero y Cliente	
<b>Objetivos asociados</b>	Aplicación Móvil para Pedidos	
<b>Requisitos asociados</b>	RF - 04 Los meseros emiten las facturas desde sus Tablets cuando los clientes piden la cuenta.	
<b>Descripción</b>	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el mesero vaya a anotar el pedido.	
<b>Precondición</b>	Sin precondición	
<b>Secuencia Normal</b>	<b>Paso</b>	<b>Acción</b>
	1	El mesero presenta el menú por medio de la tablet
	2	Se registran las solicitudes de los clientes
	3	Se calcula el presupuesto actual de la orden
	4	El mesero registra el pedido a través de la Tablet
<b>Postcondición</b>	RF - 02 Registro de actividad de cocina	
<b>Excepciones</b>	<b>Paso</b>	<b>Acción</b>
	2	En caso de que no haya un platillo del menú disponible se notificará.
<b>Rendimiento</b>	<b>Paso</b>	<b>Cota de tiempo</b>
	1	120 segundos
	2	30 segundos
	3	10 segundos
	4	10 segundos
<b>Frecuencia esperada</b>	n veces	
<b>Comentarios</b>	La frecuencia esperada podrá ser más de una vez, esto dependerá de las necesidades que presente el cliente.	

<b>RF- 02</b>	Registro de actividad de cocina	
<b>Actor Principal</b>	Cocinero	
<b>Objetivos asociados</b>	Implementación de la Pizarra Interactiva para la Optimización del proceso.	
<b>Requisitos asociados</b>	RF -01 Recepción de peticiones en las mesas RF -03 Entrega de platos	
<b>Descripción</b>	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando se muestran las peticiones en la cocina a través de una pizarra interactiva.	
<b>Precondición</b>	RF -01 Recepción de peticiones en las mesas	
<b>Secuencia Normal</b>	<b>Paso</b>	<b>Acción</b>
	1	Se muestran los platos solicitados ordenados por las mesas
	2	Preparar los platos solicitados tomando en cuenta todas las peticiones señaladas
	3	Los meseros toman los platos ya listos para ser servidos una vez los han terminado de cocinar
<b>Postcondición</b>	RF -03 Entrega de platos	
<b>Excepciones</b>	<b>Paso</b>	<b>Acción</b>
	2	En caso de un contratiempo en la preparación del pedido se le deberá indicar al mesero para que este lo notifique.
<b>Rendimiento</b>	<b>Paso</b>	<b>Cota de tiempo</b>
	1	1 minutos
	2	45 minutos
	3	5 minutos
<b>Frecuencia esperada</b>	n veces	
<b>Comentarios</b>	La frecuencia esperada podrá ser más de una vez, esto dependerá de las necesidades que se presenten.	

<b>RF-03</b>	Entrega de platos
--------------	-------------------

<b>Actor Principal</b>	Mesero y Cliente	
<b>Objetivos asociados</b>	Implementación de la Pizarra Interactiva para Optimización del proceso	
<b>Requisitos asociados</b>	RF- 01 Recepción de peticiones en las mesas RF- 02 Registro de actividad de cocina	
<b>Descripción</b>	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando se indique en la pizarra interactiva que la orden de determinada mesa ha sido preparada.	
<b>Precondición</b>	RF - 02 Registro de cocina	
<b>Secuencia Normal</b>	<b>Paso</b>	<b>Acción</b>
	1	El mesero recibe la notificación de la pizarra indicando que la orden de cierta mesa ha sido terminada
	2	El mesero recibe la orden de los clientes desde la cocina
	3	El mesero entrega la orden a la mesa indicada
<b>Postcondición</b>	RF - 01 Recepción de peticiones en las mesas RF - 04 Facturación	
<b>Excepciones</b>	<b>Paso</b>	<b>Acción</b>
	2	En caso de haber un imprevisto en la preparación de los platos se deberá notificar.
<b>Rendimiento</b>	<b>Paso</b>	<b>Cota de tiempo</b>
	1	1 segundos
	2	10 segundos
	3	30 segundos
<b>Frecuencia esperada</b>	n veces	
<b>Comentarios</b>	<p>Las postcondiciones están condicionadas a las necesidades que presenten los clientes, ya que se puede volver a realizar otro pedido o pedir la cuenta.</p> <p>La frecuencia esperada podrá ser más de una vez, esto dependerá de las necesidades que se presenten.</p>	

<b>RF- 04</b>	Facturación
<b>Actor Principal</b>	Meseros
<b>Objetivos asociados</b>	Multiples Opciones de Pago

<b>Requisitos asociados</b>	RF - 03 Entrega de platos	
<b>Descripción</b>	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando se solicite emitir una factura.	
<b>Precondición</b>	RF - 03 Entrega de platos	
<b>Secuencia Normal</b>	<b>Paso</b>	<b>Acción</b>
	1	El mesero a través de la Tablet inicia el proceso de facturación
	2	El mesero solicita los datos de facturación del cliente
	3	El mesero registra los datos en el sistema para posteriormente generarla y enviarla por correo al cliente
<b>Postcondición</b>	Ninguna postcondición	
<b>Excepciones</b>	<b>Paso</b>	<b>Acción</b>
	3	En caso de que la tablet no funcione se optará por cambiarla por una nueva tablet PC y se continuará con el proceso
<b>Rendimiento</b>	<b>Paso</b>	<b>Cota de tiempo</b>
	1	1 segundos
	2	5 segundos
	3	40 segundos
<b>Frecuencia esperada</b>	n veces	
<b>Comentarios</b>	La frecuencia esperada podrá ser más de una vez, esto dependerá de las necesidades que se presenten.	

### 3.3. Diseño detallado de las clases.

En esta sección se describe completamente las clases (nombre, atributos y métodos) CRC y sus interfaces. Se incluye la documentación de las dependencias de cada clase con otras clases y paquetes.

*Clases:*

- **Mesa:**
  - *Responsabilidades:*
    - Almacenar información sobre una mesa específica.
  - *Atributos:*
    - numeroMesa: int
    - meseroAsignado: string
    - factura: Factura.
  - *Operaciones:*
    - AsignarMesero(MeseroID)
    - RegistrarVenta(DetalleVenta)
  - *Colaboradores*
    - Paquete de Gestión de Mesas
    - Clase Factura
    - Clase Mesero
- **Mesero:**
  - *Responsabilidades:*
    - Almacenar información sobre un mesero.
  - *Atributos:*
    - dni: string
    - apellidos: string
    - nombre: string
  - *Operaciones:*
    - ConsultarVentas(MesaNumero)
  - *Colaboradores*
    - Paquete de Gestión de Mesas
    - Clase Factura
- **Peticion:**



- *Responsabilidades:*
  - Almacenar detalles de una petición de un comensal.
- *Atributos:*
  - mesaNumero: int
  - detallesPlatos: string
- *Operaciones:*
  - EnviarPetición()
  - VisualizarEnCocina()
  - AvisarFinalizaciónPlato(PlatoID)
- *Colaboradores*
  - Paquete de Peticiones y Cocina
  - Clase Plato
  - Clase Mesa
- **Plato:**
  - *Responsabilidades:*
    - Representar un plato específico.
  - *Atributos:*
    - nombre: string
    - tipo: string
    - tiempoElaboración: string
  - *Operaciones:*
    - Elaborar()
    - ListoParaServir()
  - *Colaboradores*
    - Paquete de Peticiones y Cocina
- **Factura:**
  - *Responsabilidades:*
    - Almacenar detalles de una factura.
  - *Atributos:*
    - id: int
    - mesaNumero: int

- detallesVenta: string
- *Operaciones:*
  - GenerarFactura()
  - ConsultarFactura()
- *Colaboradores*
  - Paquete de Facturación
  - Clase Mesa
- **PedidoProveedor:**
  - *Responsabilidades:*
    - Representa un pedido específico a un proveedor.
  - *Atributos:*
    - proveedorID: int
    - detallesPedido: string
  - *Operaciones:*
    - RealizarPedido()
  - *Colaboradores:*
    - Paquete de Aprovisionamiento
- **Mercancia:**
  - *Responsabilidades:*
    - Representa la mercancía recibida.
  - *Atributos:*
    - proveedorID: int
    - detallesMercancia: string
  - *Operaciones:*
    - RecibirMercancia()
  - *Colaboradores:*
    - Paquete de Aprovisionamiento
- **RegistroConsumo:**
  - *Responsabilidades:*
    - Almacenar detalles del consumo de alimentos.

- *Atributos:*
  - detalleConsumo: string
- *Operaciones:*
  - RegistrarConsumo()
- *Colaboradores:*
  - Paquete de Consumo de Alimentos
- **Autenticacion:**
  - *Responsabilidades:*
    - Validar credenciales de usuarios.
  - *Atributos:*
    - usuariosRegistrados: string
  - *Operaciones:*
    - Autenticar()
  - *Colaboradores:*
    - Paquete de Seguridad
- **ControlAcceso:**
  - *Responsabilidades:*
    - Gestionar políticas de control de acceso.
  - *Atributos:*
    - roles: string
    - recursos: string
  - *Operaciones:*
    - VerificarAcceso().
  - *Colaboradores:*
    - Paquete de Seguridad
    -

## 4. Diseño de la arquitectura de módulos del sistema

### 1. Capa de presentación:

- **Descripción:**

La capa de presentación es la interfaz directa con los usuarios finales, en particular los servidores que utilizan tabletas. Su principal responsabilidad es mostrar menús, registrar pedidos y gestionar la facturación, actuando como punto de entrada para la interacción del usuario.

- **Submódulos asignados: Ninguno asignado directamente.**

La capa de presentación interactúa principalmente con la capa de aplicación para coordinar y gestionar las operaciones.

## **2. Capa de aplicación o lógica de negocio:**

- **Descripción:**

La capa de aplicación es el corazón del sistema y alberga la lógica empresarial esencial. Contiene subsistemas especializados en funciones como gestión de mesas, solicitudes, cocina, entrega de alimentos, facturación, adquisición y consumo de alimentos. Coordina las operaciones y garantiza la coherencia de los datos.

- **Submódulos asignados:**

- 1. Submódulo de gestión de tablas:**

- Asignación de camareros a mesas.
- Registro y consulta de ventas por mesa.
- Coordinación con la capa de aplicación para actualización de datos.

- 2. Submódulo de Peticiones y Cocina:**

- Registro y visualización de solicitudes.
- Coordinación de la preparación de platos en la cocina.
- Comunicación con la capa de aplicación para la gestión de pedidos.

- 3. Submódulo de facturación:**

- Generación y gestión de facturas.
- Coordinación con la capa de aplicación para la obtención de información de pedidos.

- 4. Submódulo de aprovisionamiento:**

- Gestión de pedidos de proveedores.
- Recepción de mercancías y facturas.
- Coordinación con la Capa de Aplicación para actualización de inventarios.

- 5. Submódulo de consumo de alimentos:**

- Registro de consumo de alimentos.
- Cálculo de unidades a descontar al final del día.
- Coordinación con la capa de aplicación para mantener registros precisos.

- 6. Submódulo de seguridad:**

- Implementación de mecanismos de autenticación.
- Cifrado de datos sensibles.
- Control de acceso basado en roles.

- 7. Submódulo de comunicación:**

- Gestión de la comunicación entre la capa de presentación y la capa de aplicación.
- Gestión de solicitudes y respuestas entre dispositivos Tablet PC y el servidor central.

## **3. Capa de persistencia:**

- **Descripción:**

La capa de persistencia es responsable de almacenar y recuperar datos. Utiliza una base de datos relacional para almacenar información esencial como detalles de tablas, pedidos, proveedores, etc. La comunicación con la capa de aplicación garantiza un acceso eficiente a los datos.

- **Submódulos asignados:**

- 1. Submódulo de base de datos:**

- Almacenamiento y recuperación de datos eficiente.
- Mantener la coherencia de la base de datos.
- Implementar transacciones para garantizar la integridad de los datos.

Esta estructura modular proporciona una clara división de responsabilidades y facilita el desarrollo y mantenimiento eficiente del sistema de gestión del restaurante.

## **5. Diseño físico de datos**

### **Entidades principales:**

- 1. Mesa:** La entidad "Mesa" almacena información sobre cada mesa en el restaurante. Guarda el número de mesa, su estado (ocupada o libre) y el mesero asignado. Esta información es crucial

para gestionar la asignación de mesas a clientes y meseros.

**Atributos:**

- **número de tabla (int):** Identificación única de cada mesa en el restaurante
- **estado (varchar):** Indica si la mesa está ocupada o libre.
- **servidor asignado (varchar):** Nombre del mesero asignado a la mesa.

2. **Orden (Pedidos):** La entidad "Pedido" registra los detalles de cada pedido realizado por los comensales. Almacena un identificador único, la descripción de los platos solicitados y la hora del pedido. Facilita el seguimiento de las solicitudes de los clientes y la coordinación en la cocina.

**Atributos:**

- **Identificador del pedido (int):** Número único que identifica cada pedido.
- **detalle de los platos pedidos (varchar):** Descripción de los platos solicitados.
- **hora del pedido (timestamp):** Registro de la hora en que se realizó el pedido.

3. **Proveedor:** La entidad "Proveedor" contiene detalles sobre los proveedores de productos para el restaurante. Incluye el nombre del proveedor, información de contacto y una lista de productos suministrados. Facilita la gestión de aprovisionamiento y la relación con los proveedores.

**Atributos:**

- **Número de proveedores (varchar):** Nombre de la empresa proveedora.
- **datos de contacto (varchar):** Información de contacto del proveedor.
- **listado de productos presentados (varchar):** Información de contacto del proveedor.

4. **Inventario:** La entidad "Inventario" gestiona la cantidad y detalles de los alimentos almacenados en el restaurante. Registra la cantidad disponible, la descripción de los productos y sus fechas de vencimiento. Permite un control eficiente de los suministros y evita problemas de caducidad.

**Atributos:**

- **stock de alimentos (int):** Cantidad disponible de cada alimento en el inventario.
- **detalle del producto (varchar):** Descripción detallada de cada producto en el inventario.
- **detalle de venta (fecha):** Fecha límite de consumo de cada alimento.

5. **Factura:** La entidad "Factura" se encarga de registrar la información de las transacciones financieras asociadas a los pedidos. Incluye un número único de factura, detalles de productos y servicios facturados, el monto total y los impuestos aplicados. Es esencial para el seguimiento financiero y la contabilidad del restaurante.

**Atributos.**

- **número de factura (int):** Identificador único de cada factura generada.
- **detalles de productos y servicios (varchar):** Descripción de los ítems facturados.
- **total (decimal):** Monto total de la factura.
- **cargos (decimal):** Monto de impuestos asociados a la factura.

**Informes:**

- **Relación orden-mesa:** Un pedido está asociado a una tabla, pero una tabla puede contener varios pedidos.
- **Relación pedido-inventario:** Un pedido implica el uso de productos en stock.
- **Relación pedido-factura:** Un pedido se refleja en una única factura, pero una factura puede contener varios pedidos.
- **Relación pedido-proveedor:** Los productos de un pedido se pueden vincular a proveedores específicos.

**Consideraciones:**

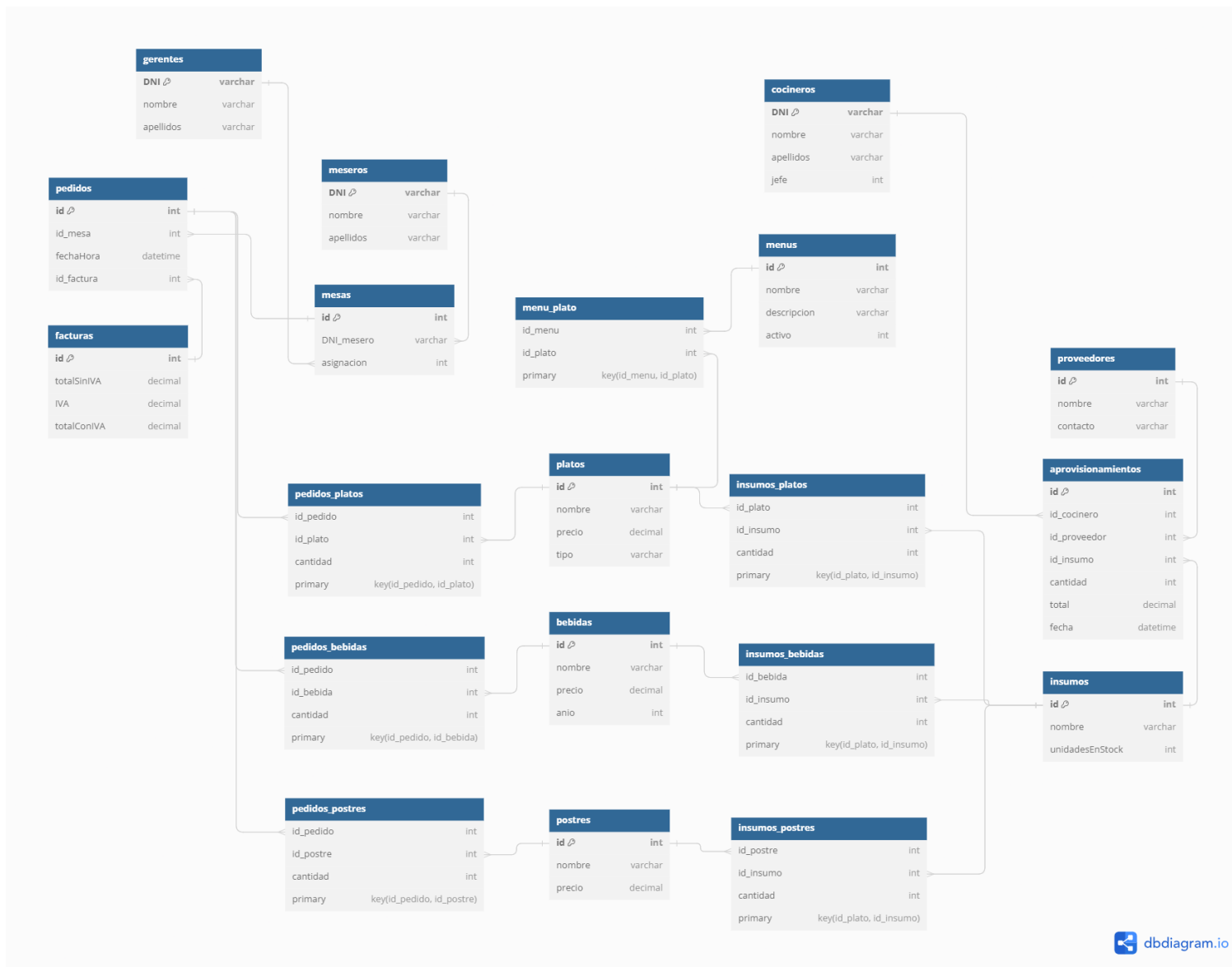
Se implementaron índices en campos clave (ID) para optimizar la velocidad de consulta. Estas restricciones de integridad referencial se implementarán para garantizar la coherencia de los datos. La base de datos está organizada de manera que permita un acceso eficiente a la información requerida por

la aplicación de capacitación.

Este diseño de datos físicos se centra en la estructura y las relaciones de entidades específicas asociadas con el submódulo de la base de datos en la capa de persistencia.

## 6. Modelo de datos y diccionario de datos

Diagrama Entidad - Relación



## Descripción de Tablas del Diagrama

### 1. Table gerentes

- DNI: Clave primaria. Identificador único para cada gerente.
- nombre: Nombre del gerente.
- apellidos: Apellidos del gerente.

### 2. Table meseros

- DNI: Clave primaria. Identificador único para cada mesero.
- nombre: Nombre del mesero.
- apellidos: Apellidos del mesero.

### 3. Table cocineros

- DNI: Clave primaria. Identificador único para cada cocinero.
- nombre: Nombre del cocinero.
- apellidos: Apellidos del cocinero.
- jefe: Indica si el cocinero es jefe de cocina (posiblemente un valor booleano o un identificador).

### 4. Table mesas

- id: Clave primaria. Identificador único para cada mesa.
- DNI\_mesero: Relación con meseros. Indica qué mesero está asignado a la mesa.
- asignacion: Relación con gerentes. Indica qué gerente ha asignado el mesero a la mesa.

### 5. Table pedidos

- id: Clave primaria. Identificador único para cada pedido.
- id\_mesa: Relación con mesas. Indica a qué mesa pertenece el pedido.
- fechaHora: Fecha y hora en que se realizó el pedido.
- id\_factura: Relación con facturas. Indica a qué factura está asociado el pedido.

### 6. Table platos

- id: Clave primaria. Identificador único para cada plato.
- nombre: Nombre del plato.
- precio: Precio del plato.
- tipo: Tipo de plato (por ejemplo, entrante, principal, etc.).

### 7. Table bebidas

- id: Clave primaria. Identificador único para cada bebida.
- nombre: Nombre de la bebida.
- precio: Precio de la bebida.
- anio: Año de la bebida, relevante para vinos y otras bebidas añejadas.

### 8. Table postres

- id: Clave primaria. Identificador único para cada postre.
- nombre: Nombre del postre.
- precio: Precio del postre.

### 9. Table pedidos\_platos

- id\_pedido: Relación con pedidos. Identifica a qué pedido pertenece.
- id\_plato: Relación con platos. Identifica qué plato se ha pedido.
- cantidad: Cantidad del plato pedido.
- primary key (id\_pedido, id\_plato): Clave primaria compuesta para identificar de forma única cada línea en la tabla.

### 10. Table pedidos\_bebidas

- id\_pedido: Relación con pedidos. Identifica a qué pedido pertenece.

- id\_bebida: Relación con bebidas. Identifica qué bebida se ha pedido.
- cantidad: Cantidad de la bebida pedida.
- primary key (id\_pedido, id\_bebida): Clave primaria compuesta.

#### 11. Table pedidos\_postres

- id\_pedido: Relación con pedidos. Identifica a qué pedido pertenece.
- id\_postre: Relación con postres. Identifica qué postre se ha pedido.
- cantidad: Cantidad del postre pedido.
- primary key (id\_pedido, id\_postre): Clave primaria compuesta.

#### 12. Table insumos

- id: Clave primaria. Identificador único para cada insumo.
- nombre: Nombre del insumo.
- unidadesEnStock: Cantidad de unidades en stock del insumo.

#### 13. Table proveedores

- id: Clave primaria. Identificador único para cada proveedor.
- nombre: Nombre del proveedor.
- contacto: Información de contacto del proveedor.

#### 14. Table aprovisionamientos

- id: Clave primaria. Identificador único para cada aprovisionamiento.
- id\_cocinero: Relación con cocineros. Identifica qué cocinero realiza el aprovisionamiento.
- id\_proveedor: Relación con proveedores. Identifica el proveedor del aprovisionamiento.
- id\_insumo: Relación con insumos. Identifica el insumo que se está aprovisionando.
- cantidad: Cantidad del insumo aprovisionado.
- total: Costo total del aprovisionamiento.
- fecha: Fecha del aprovisionamiento.

#### 15. Table facturas

- id: Clave primaria. Identificador único para cada factura.
- totalSinIVA: Total de la factura sin incluir IVA.
- IVA: Monto del IVA.
- totalConIVA: Total de la factura incluyendo IVA.

#### 16. Table menus

- id: Clave primaria. Identificador único para cada menú.
- nombre: Nombre del menú.
- descripcion: Descripción del menú.
- activo: Indica si el menú está activo o no.

#### 17. Table menu\_plato

- id\_menu: Relación con menus. Identifica a qué menú pertenece.
- id\_plato: Relación con platos. Identifica qué plato se incluye en el menú.
- primary key (id\_menu, id\_plato): Clave primaria compuesta.

#### 18. Table insumos\_platos

- id\_plato: Relación con platos. Identifica el plato.
- id\_insumo: Relación con insumos. Identifica el insumo utilizado en el plato.
- cantidad: Cantidad del insumo utilizado en el plato.
- primary key (id\_plato, id\_insumo): Clave primaria compuesta.

#### 19. Table insumos\_bebidas

- id\_bebida: Relación con bebidas. Identifica la bebida.
- id\_insumo: Relación con insumos. Identifica el insumo utilizado en la bebida.
- cantidad: Cantidad del insumo utilizado en la bebida.
- primary key (id\_bebida, id\_insumo): Clave primaria compuesta.

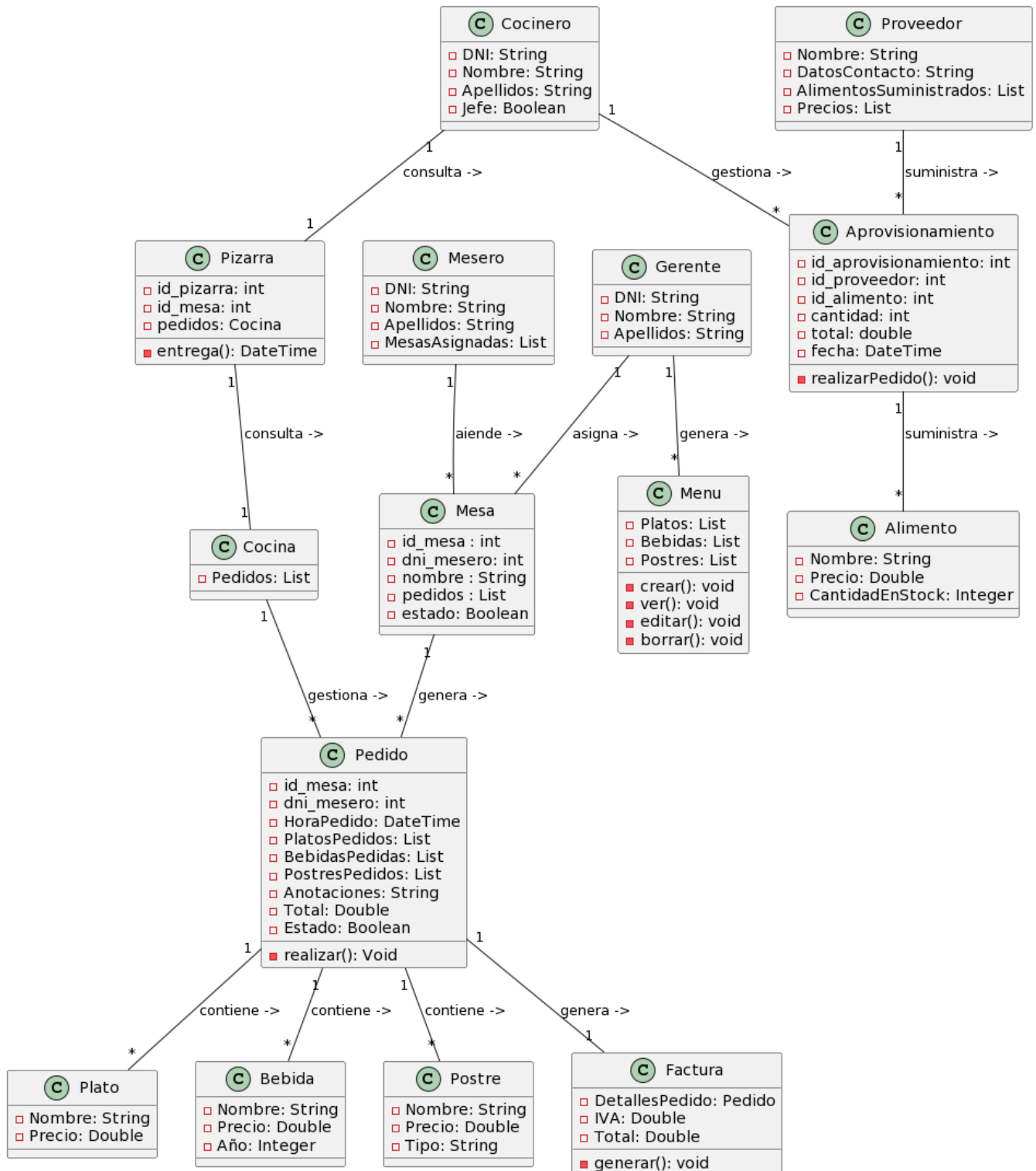
#### 20. Table insumos\_postres

- id\_postre: Relación con postres. Identifica el postre.
- id\_insumo: Relación con insumos. Identifica el insumo utilizado en el postre.
- cantidad: Cantidad del insumo utilizado en el postre.



- primary key (id\_postre, id\_insumo): Clave primaria compuesta.

## 6.1. Modelo Entidad – Relación (Diagrama de Clases)



## 6.2. Diseño de datos

Tabla 2 Diccionario de datos

Número y nombre de la entidad		TBL - <1>	gerentes				
Descripción		Almacena información sobre los gerentes del restaurante, incluyendo su identificación y datos personales.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeter minado	Auto numérico
1	DNI	Identificador único del gerente.	varchar	NO	PK	N/A	SI
2	nombre	Nombre del gerente.	varchar	NO	N/A	N/A	NO
3	apellidos	Apellidos del gerente.	varchar	NO	N/A	N/A	NO

Número y nombre de la entidad		TBL - <2>	meseros				
Descripción		Contiene los detalles de los meseros, como su identificación y nombre completo.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeter minado	Auto numérico
1	DNI	Identificador único del mesero.	varchar	NO	PK	No aplica	NO
2	nombre	Nombre del mesero.	varchar	NO		No aplica	NO
3	apellidos	Apellidos del mesero.	varchar	NO		No aplica	NO

Número y nombre de la entidad		TBL - <3>	cocineros				
Descripción		Guarda información sobre los cocineros, incluyendo su identificación, nombre, y si alguno de ellos es el jefe de cocina.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeter minado	Auto numérico
1	DNI	Identificador único del cocinero.	varchar	NO	PK	No aplica	NO
2	nombre	Nombre del cocinero.	varchar	NO		No aplica	NO
3	apellidos	Apellidos del cocinero.	varchar	NO		No aplica	NO
4	jefe	Indica si el cocinero es jefe de cocina.	int	SI		No aplica	NO

Número y nombre de la entidad		TBL - <4>	mesas				
-------------------------------	--	-----------	-------	--	--	--	--

Descripción		Registra las mesas del restaurante, su identificación, y la asignación de meseros y gerentes a cada una.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeter minado	Auto numérico
1	id	Identificador único de la mesa.	int	NO	PK	No aplica	SI
2	DNI_mesero	Relaciona la mesa con el mesero asignado.	varchar	SI	FK	No aplica	NO
3	asignacion	Relaciona la mesa con el gerente que realizó la asignación.	int	SI	FK	No aplica	N

Número y nombre de la entidad		TBL - <5>	pedidos				
Descripción		Almacena los pedidos realizados por los clientes, incluyendo la mesa, la fecha y hora del pedido, y la factura asociada.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeter minado	Auto numérico
1	id	Identificador único del pedido.	int	NO	PK	No aplica	SI
2	id_mesa	Relaciona el pedido con la mesa donde se realizó.	int	NO	FK	No aplica	NO
3	fechaHora	Fecha y hora en que se realizó el pedido.	datetime	NO		No aplica	NO
4	id_factura	Relaciona el pedido con su factura correspondiente.	int	SI	FK	No aplica	NO

Número y nombre de la entidad		TBL - <6>	platos				
Descripción		Contiene información sobre los platos disponibles en el menú, incluyendo su nombre, precio y tipo.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeter minado	Auto numérico
1	id	Identificador único del plato.	int	NO	PK	No aplica	SI
2	nombre	Nombre del plato.	varchar	NO		No aplica	NO
3	precio	Precio del plato.	decimal	NO		No aplica	NO
4	tipo	Tipo de plato (primer plato, segundo plato, etc.).	varchar	SI		No aplica	NO

Número y nombre de la entidad		TBL - <7>	bebidas				
-------------------------------	--	-----------	---------	--	--	--	--

Descripción		Almacena detalles de las bebidas disponibles, como su nombre, precio y, en el caso de los vinos, el año.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeter minado	Auto numérico
1	id	Identificador único de la bebida.	int	NO	PK	No aplica	SI
2	nombre	Nombre de la bebida.	varchar	NO		No aplica	NO
3	precio	Precio de la bebida.	decimal	NO		No aplica	NO
4	anio	Año de la bebida, relevante para vinos.	int	SI		No aplica	NO

Número y nombre de la entidad		TBL - <8>	postres				
Descripción		Guarda información sobre los postres ofrecidos, incluyendo su nombre y precio.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeter minado	Auto numérico
1	id	Identificador único del postre.	int	NO	PK	No aplica	SI
2	nombre	Nombre del postre.	varchar	NO		No aplica	NO
3	precio	Precio del postre.	decimal	NO		No aplica	NO

Número y nombre de la entidad		TBL - <9>	pedidos_platos				
Descripción		Relaciona los pedidos con los platos solicitados, incluyendo la cantidad de cada plato en un pedido específico.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeter minado	Auto numérico
1	id_pedido	Identificador del pedido.	int	NO	FK	No aplica	NO
2	id_plato	Identificador del plato solicitado en el pedido.	int	NO	FK	No aplica	NO
3	cantidad	Cantidad del plato solicitado.	int	NO		No aplica	NO

Número y nombre de la entidad		TBL - <10>	pedidos_bebidas				
-------------------------------	--	------------	-----------------	--	--	--	--

Descripción		Asocia los pedidos con las bebidas solicitadas, detallando la cantidad de cada bebida en un pedido.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeterminado	Auto numérico
1	id_pedido	Identificador del pedido.	int	NO	FK	No aplica	NO
2	id_bebida	Identificador de la bebida solicitada en el pedido.	int	NO	FK	No aplica	NO
3	cantidad	Cantidad de la bebida solicitada.	int	NO		No aplica	NO

Número y nombre de la entidad		TBL - <11>	pedidos_postres				
Descripción		Vincula los pedidos con los postres solicitados, especificando la cantidad de cada postre en un pedido.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeter minado	Auto numérico
1	id_pedido	Identificador del pedido.	int	NO	FK	No aplica	NO
2	id_postre	Identificador del postre solicitado en el pedido.	int	NO	FK	No aplica	NO
3	cantidad	Cantidad del postre solicitado.	int	NO		No aplica	NO

Número y nombre de la entidad		TBL - <12>	insumos				
Descripción		Almacena información sobre los insumos (ingredientes y productos) utilizados en el restaurante, incluyendo su stock actual.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeter minado	Auto numérico
1	id	Identificador único del insumo.	int	NO	PK	No aplica	SI
2	nombre	Nombre del insumo.	varchar	NO		No aplica	NO
3	unidadesEnStock	Cantidad de unidades del insumo en stock.	int	NO		No aplica	NO

Número y nombre de la entidad		TBL - <13>	proveedores				
-------------------------------	--	------------	-------------	--	--	--	--

Descripción		Contiene detalles de los proveedores del restaurante, incluyendo su nombre y contacto.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeterminado	Auto numérico
1	id	Identificador único del proveedor.	int	NO	PK	No aplica	SI
2	nombre	Nombre del proveedor.	varchar	NO		No aplica	NO
3	contacto	Información de contacto del proveedor.	varchar	NO		No aplica	NO

Número y nombre de la entidad		TBL - <14>	aprovisionamientos				
Descripción		Registra los aprovisionamientos de insumos realizados por los cocineros, incluyendo detalles del proveedor, insumo, cantidad y coste.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeter minado	Auto numérico
1	id	Identificador único del aprovisionamiento.	int	NO	PK	No aplica	SI
2	id_cocinero	Identificador del cocinero que realiza el aprovisionamiento.	int	NO	FK	No aplica	NO
3	id_proveedor	Identificador del proveedor del insumo.	int	NO	FK	No aplica	NO
4	id_insumo	Identificador del insumo aprovisionado.	int	NO	FK	No aplica	NO
5	cantidad	Cantidad del insumo aprovisionado.	int	NO		No aplica	NO
6	total	Costo total del aprovisionamiento.	decimal	NO		No aplica	NO
7	fecha	Fecha del aprovisionamiento.	datetime	NO		No aplica	NO

Número y nombre de la entidad	TBL - <15>	facturas
Descripción	Almacena las facturas generadas por los pedidos, incluyendo el total sin IVA, el	

		IVA y el total con IVA.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeterminado	Auto numérico
1	id	Identificador único de la factura.	int	NO	PK	No aplica	SI
2	totalSinIVA	Total de la factura sin incluir IVA.	decimal	NO		No aplica	NO
3	IVA	Monto del IVA.	decimal	NO		No aplica	NO
4	totalConIVA	Total de la factura incluyendo IVA.	decimal	NO		No aplica	NO

Número y nombre de la entidad		TBL - <16>	menus				
Descripción		Guarda información sobre los menús disponibles en el restaurante, incluyendo su nombre, descripción y si están activos.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeter minado	Auto numérico
1	id	Identificador único del menú.	int	NO	PK	No aplica	SI
2	nombre	Nombre del menú.	varchar	NO		No aplica	NO
3	descripcion	Descripción del menú.	varchar	SI		No aplica	NO
4	activo	Indica si el menú está activo o no.	boolean	NO		No aplica	NO

Número y nombre de la entidad		TBL - <17>	menú_plato				
Descripción		Relaciona los menús con los platos que incluyen, permitiendo la configuración de menús variados.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeterminado	Auto numérico
1	id_menu	Identificador del menú.	int	NO	FK	No aplica	NO
2	id_plato	Identificador del plato incluido en el menú.	int	NO	FK	No aplica	NO

Número y nombre de la entidad		TBL - <18>	insumos_platos				
Descripción		Asocia los platos con los insumos necesarios para su preparación, incluyendo la cantidad de cada insumo.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeter minado	Auto numérico
1	id_plato	Identificador del plato.	int	NO	FK	No aplica	NO
2	id_insumo	Identificador del insumo utilizado en el plato.	int	NO	FK	No aplica	NO
3	cantidad	Cantidad del insumo utilizado en el plato.	int	NO		No aplica	NO

Número y nombre de la entidad		TBL - <19>	insumos_bebidas				
Descripción		Vincula las bebidas con los insumos necesarios para su preparación o presentación.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeter minado	Auto numérico
1	id_bebida	Identificador de la bebida.	int	NO	FK	No aplica	NO
2	id_insumo	Identificador del insumo utilizado en la bebida.	int	NO	FK	No aplica	NO
3	cantidad	Cantidad del insumo utilizado en la bebida.	int	NO		No aplica	NO

Número y nombre de la entidad		TBL - <20>	insumos_postres				
Descripción		Relaciona los postres con los insumos requeridos para su elaboración, detallando la cantidad de cada insumo.					
Nro.	Nombre de la Columna	Descripción	Tipo de datos	Permitir Valores NULL	Llave	Valor Predeter minado	Auto numérico
1	id_bebida	Identificador de la bebida.	int	NO	FK	No aplica	NO
2	id_insumo	Identificador del insumo utilizado en la bebida.	int	NO	FK	No aplica	NO
3	cantidad	Cantidad del insumo utilizado en la bebida.	int	NO		No aplica	NO



## **7. Prototipo de pantallas**

### **1. Pantallas de Gestión de Mesas:**

- Asignación de Meseros:
  - Pantalla para asignar meseros a mesas.
- Registro y Consulta de Ventas:
  - Pantalla para registrar y consultar ventas por mesa.

### **2. Pantallas de Peticiones y Cocina:**

- Registro y Visualización de Peticiones:
  - Pantalla para registrar y visualizar peticiones.
- Coordinación en Cocina:
  - Pantalla que muestra la coordinación de la elaboración de platos en la cocina.

### **3. Pantallas de Facturación:**

- Generación y Gestión de Facturas:
  - Pantalla para la generación y gestión de facturas.

### **4. Pantallas de Aprovisionamiento:**

- Gestión de Pedidos a Proveedores:
  - Pantalla para gestionar pedidos a proveedores.
- Recepción de Mercancía y Facturas:
  - Pantalla para la recepción de mercancía y facturas.

### **5. Pantallas de Consumo de Alimentos:**

- Registro del Consumo de Alimentos:
  - Pantalla para registrar el consumo de alimentos.
- Cálculo de Unidades a Descontar:
  - Pantalla para el cálculo de unidades a descontar al final del día.

### **6. Pantallas de Seguridad:**

- Mecanismos de Autenticación:
  - Pantalla para la implementación de mecanismos de autenticación.
- Control de Acceso basado en Roles:
  - Pantalla para gestionar el control de acceso basado en roles.

### **7. Pantallas de Comunicación:**

- Gestión de Comunicación:
  - Pantalla que muestra la gestión de la comunicación entre la Capa de Presentación y la Capa de Aplicación.
- Manejo de Solicitudes y Respuestas:
  - Pantalla para el manejo de solicitudes y respuestas entre dispositivos Tablet PC y el servidor central.

### **8. Pantallas de Base de Datos:**

- Almacenamiento y Recuperación de Datos:
  - Pantalla para la gestión eficiente del almacenamiento y recuperación de datos.
- Mantenimiento de la Consistencia:
  - Pantalla para el mantenimiento de la consistencia de la base de datos.
- Implementación de Transacciones:
  - Pantalla que refleje la implementación de transacciones para garantizar la integridad de los datos.

## Mapa de Navegación Web



## Mapa de Navegación Web

