**Nombre proyecto: Sistema de Gestión Integral para Restaurantes"**

### Objetivo del Software

Desarrollar un sistema de gestión para restaurantes que facilite la administración de pedidos, facturación, reservaciones, y visualización del menú tanto para empleados como para clientes de manera eficiente.

### Metodología de Desarrollo

**Metodología Ágil (Scrum)**: Esta metodología se centra en iteraciones cortas y entregables incrementales, lo que permite una mejora continua del producto y la incorporación de feedback constante de los usuarios. Ideal para proyectos académicos, ya que permite manejar cambios en los requerimientos y aprender de los errores en el proceso de desarrollo.

### Lista de Requerimientos o Historias de Usuarios

|  |  |
| --- | --- |
| Historia No. 1 **Como:** Empleado  **Quiero:** Poder facturar los pedidos realizados a clientes en el restaurante.  **Para poder:** Mantener un registro adecuado de las transacciones.  **Prioridad:** Alta | Historia No. 2 **Como:** Empleado  **Quiero:** Generar pedidos para los clientes que los realicen en persona.  **Para poder:** Asegurar que la cocina reciba las solicitudes correctamente.  **Prioridad:** Alta |
| Historia No. 3 **Como:** Cliente (Domicilio)  **Quiero:** Hacer un pedido a través de la página web del restaurante.  **Para poder:** Recibir mi comida a domicilio de manera conveniente.  **Prioridad:** Alta | Historia No. 4 **Como:** Cliente (Domicilio)  **Quiero:** Hacer una reservación en el restaurante a través de la página web.  **Para poder:** Asegurar mi mesa con anticipación.  **Prioridad:** Media |
| Historia No. 5 **Como:** Empleado  **Quiero:** Gestionar las reservaciones realizadas por los clientes.  **Para poder:** Garantizar la disponibilidad de mesas en el restaurante.  **Prioridad:** Media | Historia No. 6 **Como:** Cliente (Domicilio) o Empleado  **Quiero:** Ver el menú del restaurante actualizado.  **Para poder:** Seleccionar correctamente los platos.  **Prioridad:** Alta |
| Historia No. 7 **Como:** Cocinero  **Quiero:** Ver una lista de pedidos por orden de llegada.  **Para poder:** Marcar cada pedido como "en proceso" o "terminado" y gestionar mejor el flujo de trabajo en la cocina.  **Prioridad:** Alta | Historia No. 8 **Como:** Mesero  **Quiero:** Ver el estado de los pedidos.  **Para poder:** Informar a los clientes y coordinar la entrega de platos terminados.  **Prioridad:** Media |

### Prototipo de la Aplicación

#### Formularios de Ingreso

1. **Formulario de Ingreso para Empleados**: Permite a los empleados ingresar con sus credenciales para acceder a sus respectivas funcionalidades.
2. **Formulario de Ingreso para Clientes**: Opción de ingreso como invitado o con cuenta registrada para clientes recurrentes.

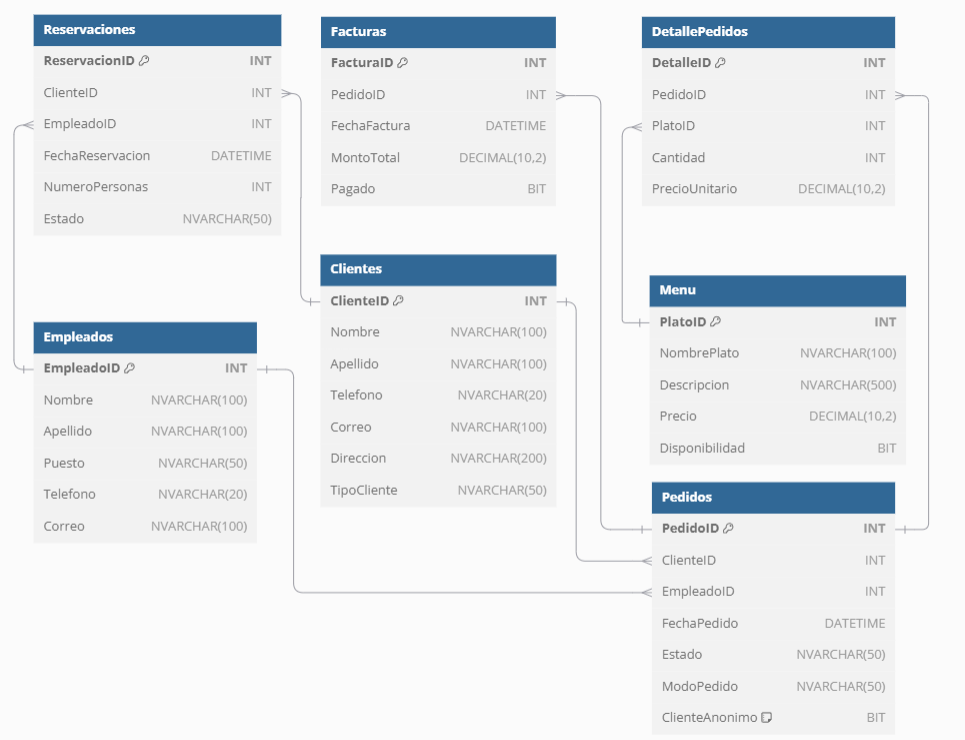
#### Menú Principal

1. **Vista para Empleados**: Opciones para visualización del menú, gestión de pedidos
2. facturación, y reservaciones.
3. **Vista para Clientes (Modo Domicilio)**: Opciones para hacer pedidos, realizar reservaciones y ver el menú.

#### Vista de Pedidos para Cocina

1. Una lista dinámica que muestra todos los pedidos entrantes con opciones para marcar como "en proceso" o "terminado".

|  |
| --- |
| -- Tabla Clientes (para clientes registrados)  CREATE TABLE Clientes (  ClienteID INT PRIMARY KEY IDENTITY(1,1),  Nombre NVARCHAR(100),  Apellido NVARCHAR(100),  Telefono NVARCHAR(20),  Correo NVARCHAR(100),  Direccion NVARCHAR(200),  TipoCliente NVARCHAR(50) -- 'Domicilio' o 'Reservación'  );  GO  -- Tabla Empleados (para empleados como meseros y cocineros)  CREATE TABLE Empleados (  EmpleadoID INT PRIMARY KEY IDENTITY(1,1),  Nombre NVARCHAR(100),  Apellido NVARCHAR(100),  Puesto NVARCHAR(50), -- 'Mesero', 'Cocinero', etc.  Telefono NVARCHAR(20),  Correo NVARCHAR(100)  );  GO  -- Tabla Menú (platos disponibles en el restaurante)  CREATE TABLE Menu (  PlatoID INT PRIMARY KEY IDENTITY(1,1),  NombrePlato NVARCHAR(100),  Descripcion NVARCHAR(500),  Precio DECIMAL(10, 2),  Disponibilidad BIT  );  GO  -- Tabla Pedidos (puede ser realizado por un cliente registrado o no)  CREATE TABLE Pedidos (  PedidoID INT PRIMARY KEY IDENTITY(1,1),  ClienteID INT NULL FOREIGN KEY REFERENCES Clientes(ClienteID), -- Puede ser NULL para clientes no registrados  EmpleadoID INT FOREIGN KEY REFERENCES Empleados(EmpleadoID),  FechaPedido DATETIME,  Estado NVARCHAR(50), -- 'Pendiente', 'En Proceso', 'Terminado'  ModoPedido NVARCHAR(50), -- 'En Restaurante', 'Domicilio'  ClienteAnonimo BIT DEFAULT 0 -- 0: Cliente registrado, 1: Cliente anónimo (no registrado)  );  GO  -- Tabla DetallePedidos (detalles de lo que se pidió: platos y cantidades)  CREATE TABLE DetallePedidos (  DetalleID INT PRIMARY KEY IDENTITY(1,1),  PedidoID INT FOREIGN KEY REFERENCES Pedidos(PedidoID),  PlatoID INT FOREIGN KEY REFERENCES Menu(PlatoID),  Cantidad INT,  PrecioUnitario DECIMAL(10, 2)  );  GO  -- Tabla Facturas (registro de facturación de pedidos)  CREATE TABLE Facturas (  FacturaID INT PRIMARY KEY IDENTITY(1,1),  PedidoID INT FOREIGN KEY REFERENCES Pedidos(PedidoID),  FechaFactura DATETIME,  MontoTotal DECIMAL(10, 2),  Pagado BIT  );  GO  -- Tabla Reservaciones (reservaciones realizadas por clientes)  CREATE TABLE Reservaciones (  ReservacionID INT PRIMARY KEY IDENTITY(1,1),  ClienteID INT NULL FOREIGN KEY REFERENCES Clientes(ClienteID), -- NULL si es cliente anónimo  EmpleadoID INT FOREIGN KEY REFERENCES Empleados(EmpleadoID), -- Siempre gestionado por empleados  FechaReservacion DATETIME,  NumeroPersonas INT,  Estado NVARCHAR(50) -- 'Pendiente', 'Confirmada', 'Cancelada'  );  GO |

****

**Explicación del Diseño:**

1. **Clientes Registrados**: Cuando un cliente hace un pedido a domicilio o una reservación, se registra en la tabla Clientes. Estos tendrán un ClienteID.
2. **Clientes No Registrados**: Para los clientes que llegan al restaurante sin haber hecho reserva, su pedido se marca con ClienteAnonimo = 1 y no tiene un ClienteID.
3. **Pedidos**:
   * **Pedidos en Restaurante**: Estos son gestionados por empleados, y se asocia un EmpleadoID al pedido.
   * **Pedidos a Domicilio**: Estos son gestionados por los clientes registrados, por lo que EmpleadoID es NULL y se usa el ClienteID del cliente que realizó el pedido.
4. **Facturación**: Se registran todas las facturas con la referencia al pedido.
5. **Reservaciones**: Todas las reservaciones son gestionadas por los empleados, pero pueden haber sido hechas por clientes registrados o anónimos.

**Flujo de Datos:**

* Un cliente puede estar registrado (con un ClienteID) o ser anónimo (ClienteAnonimo = 1).
* Los pedidos pueden ser gestionados por empleados (si son en restaurante) o por clientes (si son a domicilio).
* La facturación y estado de los pedidos son administrados en función de los empleados o clientes que interactúan con el sistema.