

Análisis de Requerimientos

Requerimientos para Consumidores

1. **Requerimiento:** Acceso a menús digitales organizados por categorías.
 - **Descripción técnica:**
 - La base de datos debe contener una tabla de **menús** asociada a **restaurantes** con categorías (ej., entradas, platos principales).
 - API RESTful para listar los menús filtrados por categoría y restaurante.
 - Validaciones para evitar menús duplicados.
 - **Restricciones:**
 - El menú debe ser accesible incluso en conexiones lentas (optimización de imágenes).
 - **Dependencias:**
 - Relación entre las tablas de **restaurantes** y **menús**.
 2. **Requerimiento:** Realizar pedidos.
 - **Descripción técnica:**
 - Endpoint en el backend para recibir los datos del pedido (usuario, ítems, total, tipo de pedido).
 - Tabla de **pedidos** en la base de datos que guarde el estado del pedido (enviado, preparando, entregado).
 - Notificaciones push o correo para confirmar el pedido al usuario.
 - **Restricciones:**
 - Validar disponibilidad del platillo antes de confirmar el pedido.
 - **Dependencias:**
 - Relación entre **pedidos**, **usuarios** y **restaurantes**.
 3. **Requerimiento:** Seguimiento del estado del pedido.
 - **Descripción técnica:**
 - Implementar un WebSocket para actualizaciones en tiempo real del estado del pedido.
 - Endpoint para consultar el estado del pedido (HTTP GET).
 - **Restricciones:**
 - Tiempo de respuesta del servidor < 1s para actualizaciones en tiempo real.
 - **Dependencias:**
 - Sincronización entre el backend y la aplicación del restaurante.
-

Requerimientos para Restaurantes

1. **Requerimiento:** Gestión de menús (crear, editar, eliminar).
 - **Descripción técnica:**
 - Endpoint en el backend con operaciones CRUD para la tabla **menús**.
 - Validaciones para campos obligatorios (nombre, precio, categoría).

- Control de acceso mediante autenticación JWT (solo administradores del restaurante pueden modificar el menú).
 - **Restricciones:**
 - Evitar menús incompletos o sin imagen.
 - **Dependencias:**
 - Relación directa con la tabla **restaurantes**.
 - 2. **Requerimiento:** Visualización en tiempo real de pedidos.
 - **Descripción técnica:**
 - Uso de WebSocket para recibir actualizaciones instantáneas de pedidos.
 - Panel de control con filtros para ver pedidos por estado (pendiente, en preparación, completado).
 - **Restricciones:**
 - Actualización en tiempo real sin recargar la página.
 - **Dependencias:**
 - Integración con los estados de los pedidos en la base de datos.
 - 3. **Requerimiento:** Estadísticas de ventas.
 - **Descripción técnica:**
 - Endpoint para consultar métricas (ventas totales, platillos más vendidos).
 - Uso de librerías gráficas (ej., Chart.js) para mostrar estadísticas visuales.
 - **Restricciones:**
 - Consultas eficientes que no impacten el rendimiento de la base de datos.
 - **Dependencias:**
 - Acceso a datos históricos de pedidos.
-

Requerimientos No Funcionales

1. **Requerimiento:** Base de datos accesible y fácil de manipular.
 - **Descripción técnica:**
 - Usar PostgreSQL para consultas estructuradas o MongoDB para flexibilidad.
 - Normalización para evitar redundancia de datos.
 - Herramientas de administración como pgAdmin o MongoDB Compass.
2. **Requerimiento:** Rendimiento.
 - **Descripción técnica:**
 - Backend debe manejar hasta 1000 solicitudes por minuto.
 - Uso de herramientas de cacheo como Redis para consultas frecuentes (menús, restaurantes).
 - **Restricciones:**
 - Latencia < 200ms en peticiones críticas.
3. **Requerimiento:** Seguridad.
 - **Descripción técnica:**

- Cifrado de contraseñas con bcrypt.
- Autenticación basada en tokens JWT.
- Certificados SSL para todas las comunicaciones.