Imagen que contiene dibujo, señal, reloj

El contenido generado por IA puede ser incorrecto.

**Facultad de Ingeniería**

Ingeniería de Sistemas e informática

**Curso**

Integrador I: Sistemas Software

**Proyecto**

“Desarrollo de un sistema web de gestión de pedidos y delivery saludable para el restaurante sabores los Huayachos en Chachapoyas”

**Integrantes**

Amaya Campos Jose Enrique - U21217065

Loyaga Lapiz Kevin - U23252108

Meza Narvaez Aaron Emanuel - U22228035

Santur Jaramillo Irvin Lois - U23212295

**Docente**

Valverde Pardave Jhonny Edgard

**Sección**

43838

Setiembre del 2025

**Chachapoyas, Lima y Chiclayo- Perú**

INDICE

[1. PRESENTACIÓN DEL CASO PROPUESTO 1](#_Toc210084113)

[2. CAPÍTULO I: Planificación del negocio y análisis del caso 2](#_Toc210084114)

[2.1. Modelo lean Canvas del caso propuesto 2](#_Toc210084115)

[2.2. El proyecto del caso propuesto 3](#_Toc210084116)

[**2.2.1.** **Project Charter del proyecto** 3](#_Toc210084117)

[**2.2.2.** **Desglose de la estructura de tareas WBS** 3](#_Toc210084118)

[2.3. Diagrama de Gantt 4](#_Toc210084119)

[2.4. Requerimientos funcionales y requerimientos no funcionales. 5](#_Toc210084120)

[**2.4.1.** **Requerimientos Funcionales** 5](#_Toc210084121)

[**2.4.2.** **Requerimientos No Funcionales** 6](#_Toc210084122)

[3. CAPÍTULO 2: Procesos y diseño de la solución 7](#_Toc210084123)

[3.1. BPMN - DIAGRAMAS DE PROCESO DE NEGOCIO: 7](#_Toc210084124)

[**3.1.1.** **Descripción del Proceso – Caso “Gestión de Despachos”** 7](#_Toc210084125)

[**3.1.2.** **Diagrama BPMN – AS-IS** 8](#_Toc210084126)

[**3.1.3.** **Diagrama BPMN – TO-BE** 10](#_Toc210084127)

[3.2. DISEÑO DE BASE DE DATOS: 12](#_Toc210084128)

[**3.2.1.** **Diagrama de casos de uso del sistema de ventas en línea “Sabores Los Huayachos”** 12](#_Toc210084129)

[**3.2.2.** **Refinamiento de RF y RNF en base a los CU** 14](#_Toc210084130)

[**3.2.3.** **Modelo lógico de la Base de Datos** 14](#_Toc210084131)

[3.3. Modelo Físico de la Base de Datos 22](#_Toc210084132)

[**3.3.1.** **Indicar el SGBD, el fabricante, la versión y distribución** 22](#_Toc210084133)

[**3.3.2.** **Detalle de cada tabla** 23](#_Toc210084134)

[**3.3.3.** **Diagrama físico** 31](#_Toc210084135)

[**3.3.4.** **Diccionario de datos** 31](#_Toc210084136)

[**3.3.5.** **Scripts del proceso de construcción de la BBDD** 37](#_Toc210084137)

[**3.3.6.** **Copias de respaldo (pendiente)** 37](#_Toc210084138)

[3.4. UI / UX 38](#_Toc210084139)

# PRESENTACIÓN DEL CASO PROPUESTO

El presente proyecto tiene como finalidad resolver las limitaciones actuales que enfrenta el restaurante **Sabores Los Huayachos**, ubicado en la ciudad de Chachapoyas, región Amazonas, el cual se especializa en la venta de comidas saludables y dietéticas mediante servicio de delivery.

En la actualidad, el negocio gestiona sus pedidos de manera manual a través de llamadas telefónicas, lo que genera diversos problemas: pérdida de información, cruces de pedidos, retrasos en la atención y dificultades para manejar múltiples solicitudes en simultáneo. Esta situación afecta directamente la satisfacción del cliente, limita la capacidad de crecimiento del restaurante y expone al negocio a una pérdida de competitividad en el mercado local.

Frente a esta problemática, se propone el desarrollo de un sistema web integral que permita digitalizar y automatizar la gestión de pedidos, pagos y entregas, incorporando módulos de catálogo de productos, carrito de compras, promociones, blog de nutrición, así como paneles diferenciados para clientes, administradores y repartidores.

El proyecto se desarrollará utilizando Java con Spring Boot como backend, HTML5, CSS3, JavaScript y Bootstrap en el frontend, y MySQL como sistema de gestión de base de datos, bajo el patrón de arquitectura MVC. La propuesta no solo busca modernizar el proceso de ventas y delivery, sino también fortalecer la presencia digital del restaurante, ampliar su cartera de clientes y posicionarse como un referente en la oferta gastronómica saludable de la región.

En síntesis, este sistema se convierte en una respuesta tecnológica y estratégica para superar las limitaciones operativas actuales, optimizar recursos, fidelizar clientes y asegurar el crecimiento sostenido de “Sabores Los Huayachos”.

# CAPÍTULO I: Planificación del negocio y análisis del caso

## Modelo lean Canvas del caso propuesto

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SOCIOS CLAVE**   * Proveedores de insumos saludables (frutas, verduras, granos) * Servicios de hosting y mantenimiento web. * Plataformas de pago (Yape, Plin, bancos). * Aliados estratégicos: gimnasios, nutricionistas, influencers locales. * Posibles convenios con empresas para menús corporativos. | **ACTIVIDADES CLAVE**   * Preparación de alimentos saludables de calidad. * Gestión de pedidos en línea y delivery. * Mantenimiento y actualización de la página web. * Creación de contenido en el blog y redes sociales. * Control de calidad y seguimiento de satisfacción del cliente. | **PROPUESTA DE VALOR**   * Venta de comida saludable y dietética con delivery rápido y confiable. * Plataforma web moderna que permite pedidos en línea sin necesidad de llamadas. * Experiencia de compra intuitiva, segura y accesible desde cualquier dispositivo. * Catálogo con información nutricional y promociones dinámicas. * Educación y fidelización a través de un blog de nutrición y estilo de vida saludable. | | **RELACIONES CON CLIENTES**   * Atención personalizada vía chat en línea o WhatsApp. * Confirmaciones y seguimiento de pedidos en tiempo real. * Sistema de testimonios y reseñas en la web. * Programas de promociones y descuentos. * En el futuro: cupones y puntos de fidelidad. | **SEGMENTOS DE CLIENTES**   * Jóvenes y adultos que buscan alimentación saludable en Chachapoyas y alrededores. * Profesionales y estudiantes con poco tiempo, que valoran el delivery rápido. * Personas con interés en dietas específicas (fitness, vegetarianas, veganas). * Empresas locales que requieran servicio de alimentación saludable para su personal. |
| **RECURSOS CLAVE**   * Cocina equipada y proveedores de insumos saludables. * Plataforma web desarrollada con PHP, MySQL, MVC, Bootstrap. * Personal: cocineros, repartidores, administrador digital. * Marca y reputación basada en confianza y salud. * Estrategia digital (redes sociales, posicionamiento SEO, marketing de contenidos). | **CANALES**   * Página web (principal canal de ventas). * Redes sociales (Facebook, Instagram, WhatsApp Business). * Delivery directo con repartidores propios. * Posibles alianzas con apps de delivery locales o nacionales**.** |
| **ESTRUCTURA DE COSTOS**   * Desarrollo y mantenimiento de la página web. * Sueldos del personal (cocina, administración, delivery). * Compra de insumos saludables. * Marketing digital (publicidad en redes sociales, SEO). * Costos logísticos (transporte, empaques, combustible). * Servicios básicos (luz, agua, internet, hosting). | | | **FLUJOS DE INGRESO**   * Venta directa de productos saludables (platos, bebidas, snacks). * Promociones y combos especiales. * Servicio de suscripción semanal o mensual de menús saludables. * Posibles ingresos por publicidad o colaboraciones en el blog (marcas de nutrición, gimnasios). | | |

## El proyecto del caso propuesto

El presente proyecto dispone de 3 estructuras.

### **Project Charter del proyecto**

Se adjunta el Project Charter en el documento **021\_ProjectCharter.docx**

### **Desglose de la estructura de tareas WBS**

**Diagrama

El contenido generado por IA puede ser incorrecto.**

## Diagrama de Gantt

## Requerimientos funcionales y requerimientos no funcionales.

### **Requerimientos Funcionales**

Son las siguientes funcionalidades específicas que el sistema debe cumplir para responder a las necesidades del negocio:

1. Gestión de usuarios:
   * Registro e inicio de sesión para clientes, administradores, repartidores y empleados.
   * Roles y permisos diferenciados según el tipo de usuario.
2. Catálogo de productos:
   * Visualización de productos organizados por categorías.
   * Filtros de búsqueda (nombre, categoría, promociones).
   * Visualización de información nutricional y precios.
3. Carrito de compras y pedidos:
   * Agregar, eliminar y modificar cantidades de productos.
   * Confirmación de pedidos con cálculo automático de subtotales y totales.
   * Seguimiento del estado del pedido (pendiente, en preparación, en camino, entregado).
4. Gestión de pedidos (Backoffice):
   * Panel administrativo para la gestión de pedidos en tiempo real.
   * Asignación de repartidores a cada pedido.
   * Visualización de historial de pedidos por cliente.
5. Gestión de promociones y descuentos:
   * Creación, activación y desactivación de promociones.
   * Aplicación de descuentos automáticos en el carrito.
6. Módulo de testimonios y blog:
   * Registro de comentarios de clientes con validación previa.
   * Publicación de artículos relacionados con nutrición y vida saludable.
7. Gestión de repartidores:
   * Panel con listado de entregas asignadas.
   * Confirmación de entrega realizada.
   * Historial de entregas completadas.
8. Gestión administrativa general:
   * CRUD de productos, usuarios, categorías, promociones y artículos del blog.
   * Generación de reportes básicos (pedidos, productos más vendidos, ingresos).

### **Requerimientos No Funcionales**

Son las siguientes características de calidad que debe cumplir el sistema para garantizar su rendimiento y confiabilidad:

1. Usabilidad:
   * Interfaz intuitiva y responsive, accesible desde dispositivos móviles, tablets y PC.
   * Formularios con validaciones en tiempo real.
2. Rendimiento:
   * Respuesta de carga de páginas menor a 3 segundos en condiciones normales.
   * Capacidad para gestionar hasta 100 usuarios concurrentes sin degradación significativa.
3. Seguridad:
   * Autenticación y autorización con Spring Security.
   * Encriptación de contraseñas con bcrypt.
   * Prevención contra ataques comunes (SQL Injection, CSRF, XSS).
4. Confiabilidad:
   * Disponibilidad del sistema ≥ 95%.
   * Respaldo automático de la base de datos diario.
5. Escalabilidad:
   * Arquitectura modular bajo el patrón MVC, con posibilidad de migrar a microservicios.
   * Base de datos normalizada y lista para ampliación de funcionalidades futuras.
6. Compatibilidad:
   * Navegadores compatibles: Chrome, Firefox, Edge, Safari.
   * Integración con sistemas de pago locales (Yape, Plin, tarjetas VISA/MasterCard) en fases posteriores.
7. Mantenibilidad:
   * Código documentado y estructurado en capas (Modelo, Vista, Controlador).
   * Soporte para pruebas unitarias y de integración con JUnit.

# CAPÍTULO 2: Procesos y diseño de la solución

## BPMN - DIAGRAMAS DE PROCESO DE NEGOCIO:

### **Descripción del Proceso – Caso “Gestión de Despachos”**

El restaurante Sabores Los Huayachos, ubicado en Chachapoyas, se dedica a la venta de comidas saludables mediante delivery. Actualmente, la gestión de pedidos, pagos y entregas se realiza de manera manual, lo cual genera problemas de pérdida de información, retrasos y falta de control.

El proceso principal corresponde a la Gestión de Despachos, que incluye:

* La recepción de pedidos por diferentes canales.
* La preparación de productos.
* La asignación y despacho por parte del repartidor.
* El control y validación de pagos.

**Se identifican dos escenarios:**

1. AS-IS (proceso actual): Gestión manual mediante llamadas, WhatsApp o atención presencial.
2. TO-BE (proceso propuesto): Gestión digital mediante un sistema web integrado que automatiza todo el ciclo de venta y despacho.

### **Diagrama BPMN – AS-IS**

**Descripción del proceso actual (AS-IS):**

* Cliente solicita pedido mediante llamada telefónica, WhatsApp o presencialmente.
* Administrador/Recepcionista registra manualmente el pedido en papel y confirma verbalmente el pedido y el monto.
* Si el pedido es presencial, cobra en caja; si es telefónico o WhatsApp, se deja para pago contra entrega.
* Cocina recibe el pedido en nota de papel, verifica insumos, prepara y empaqueta.
* Repartidor recibe el pedido anotado, lo entrega al cliente y cobra en efectivo (contra entrega).
* Los pagos son verificados manualmente al regresar el repartidor o mediante confirmación administrativa.

**Problemas:**

* Registros en papel → alta probabilidad de pérdida.
* Falta de trazabilidad de estados del pedido.
* Cobros descentralizados → riesgo de errores.
* Comunicación lenta entre administrador, cocina y repartidor.

**El diagrama AS-IS se representa con pools y lanes:**

* Pool Cliente.
* Pool Restaurante con lanes: Administrador/Recepcionista, Cocina, Repartidor.
* Flujo de tareas manuales y gateways (tipo de pedido, tipo de pago

Diagrama

El contenido generado por IA puede ser incorrecto.**Diagrama BPMN – AS-IS**

### **Diagrama BPMN – TO-BE**

**Descripción del proceso propuesto (TO-BE):**

Con la intervención del equipo informático y la implementación del sistema web en Spring Boot + MySQL, se propone digitalizar el flujo:

* Cliente accede a la web, visualiza productos, promociones y blog.
* Agrega productos al carrito, inicia sesión o se registra, define dirección de entrega validada en el sistema.
* Selecciona método de pago (Yape, Plin, tarjeta, transferencia, contra entrega) y el tipo de comprobante (boleta/factura).
* El Sistema Web registra el pedido, valida stock y cobertura, notifica al administrador y actualiza estados en la cuenta del cliente.
* Administrador/Recepcionista valida el pago (si es en línea), genera comprobante electrónico, asigna repartidor y coordina con cocina.
* Cocina recibe el pedido digital, prepara y marca como “listo para enviar”.
* Repartidor recibe notificación en su panel, visualiza detalle del pedido, actualiza estados (“en camino”, “entregado”), y valida pagos contra entrega.
* Todo el proceso queda registrado en el sistema, con trazabilidad completa y dashboard para reportes.

**Ventajas del TO-BE:**

* Centralización digital → elimina registros en papel.
* Validación automática de stock y cobertura de reparto.
* Control en tiempo real de pedidos y pagos.
* Seguimiento de pedidos desde la cuenta del cliente.
* Reportes automáticos para decisiones estratégicas.

**El diagrama TO-BE incluye:**

* Pool Cliente.
* Pool Sistema Web.
* Pool Restaurante con lanes: Administrador/Recepcionista, Cocina, Repartidor.
* Gateways: tipo de cuenta, validación de dirección, disponibilidad de stock, tipo de pago.

**Diagrama

El contenido generado por IA puede ser incorrecto.**

## DISEÑO DE BASE DE DATOS:

### **Diagrama de casos de uso del sistema de ventas en línea “Sabores Los Huayachos”**

**Actores**

* Cliente
* Administrador
* Recepcionista
* Repartidor

**Procesos principales (Casos de Uso – CU)**

**Cliente**

* CU01: Explorar catálogo de productos
* CU02: Gestionar cuenta y direcciones
* CU03: Agregar productos al carrito
* CU04: Confirmar pedido
  + «include» Seleccionar método de pago
  + «include» Seleccionar comprobante (boleta/factura)
* CU05: Consultar estado de pedido
* CU06: Registrar testimonio / reseña
* CU07: Enviar consulta (contacto)

**Recepcionista**

* CU08: Validar pedido recibido
  + «include» Validar pago (en línea / contraentrega)
  + «include» Generar comprobante
* CU09: Asignar repartidor según zona
* CU10: Actualizar estados del pedido

**Repartidor**

* CU11: Visualizar pedidos asignados
  + «include» Cambiar estado a “En camino / Entregado / Cancelado”
* CU12: Validar pago contra entrega

**Administrador**

* CU13: Gestionar productos, categorías y promociones
* CU14: Gestionar usuarios (repartidores y recepcionistas)
* CU15: Gestionar blog de nutrición
* CU16: Visualizar reportes en dashboard

**Diagrama de casos de uso del sistema**

**Diagrama

El contenido generado por IA puede ser incorrecto.**

### **Refinamiento de RF y RNF en base a los CU**

**Requisitos Funcionales (RF)**

**RF1.Gestión de clientes y usuarios**

El sistema permitirá gestionar toda la información relacionada con clientes y usuarios internos del restaurante.

* RF1.01. Registro de clientes: se almacenará código de cliente, nombres, apellidos, correo, teléfono y contraseña encriptada.
* RF1.02. Direcciones de clientes: se permitirá registrar múltiples direcciones, asociadas a zonas de reparto y provincias válidas.
* RF1.03. Gestión de usuarios internos: se registrará administrador, recepcionista y repartidor, cada uno con rol, correo, contraseña y estado de cuenta.
* RF1.04. El administrador podrá habilitar o deshabilitar usuarios internos y resetear contraseñas.

**RF2. Gestión de productos y catálogo**

El sistema permitirá administrar el catálogo de productos ofrecidos.

* RF2.01. Se almacenará código de producto, nombre, descripción, precio unitario, descuento, imagen y stock disponible.
* RF2.02. Cada producto estará asociado a una categoría activa.
* RF2.03. El administrador podrá crear, modificar y desactivar productos y categorías.
* RF2.04. El cliente podrá visualizar catálogo con fotos, promociones activas y disponibilidad en tiempo real.

**RF3. Procesos de compra (pedidos)**

El sistema gestionará los pedidos realizados por los clientes en línea.

* RF3.01. Carrito de compras: el cliente podrá agregar, modificar y eliminar productos antes de confirmar el pedido.
* RF3.02. Confirmación del pedido: el cliente deberá iniciar sesión o registrarse antes de finalizar la compra.
* RF3.03. Validación de zona de entrega: el sistema verificará si la dirección ingresada pertenece a una zona y provincia atendida.
* RF3.04. El sistema generará automáticamente el código de pedido, sucursal asignada y fecha/hora de creación.
* RF3.05. Se almacenará detalle del pedido con productos, cantidades, precios unitarios y descuentos.

**RF4. Gestión de pagos**

El sistema permitirá registrar y validar pagos realizados por los clientes.

* RF4.01. Se admitirán pagos en línea (Yape, Plin, transferencia, tarjeta) y contra entrega.
* RF4.02. El sistema permitirá pagos parciales hasta cubrir el monto total del pedido.
* RF4.03. Se registrará código de pago, pedido asociado, método de pago, monto, moneda, fecha y referencia.
* RF4.04. El recepcionista validará pagos en línea verificando referencias (voucher).
* RF4.05. Los pagos rechazados/anulados cambiarán el estado del pedido a “pendiente de pago” y notificarán al cliente.
* RF4.06. Los pagos contra entrega quedarán bajo responsabilidad del repartidor y deberán ser confirmados en el sistema.

**RF5. Comprobantes electrónicos**

El sistema generará comprobantes de pago según la normativa tributaria.

* RF5.01. Se podrá emitir boletas y facturas, asociadas a un pedido.
* RF5.02. Cada comprobante incluirá serie, número, fecha, moneda, subtotal, IGV y total.
* RF5.03. Los comprobantes quedarán almacenados en formato digital (PDF/XML).
* RF5.04. El sistema permitirá integración con SUNAT u otro sistema externo de validación de comprobantes.

**RF6. Gestión de entregas y repartidores**

El sistema permitirá asignar y controlar las entregas.

* RF6.01. El recepcionista asignará repartidor según la zona y sucursal.
* RF6.02. El repartidor recibirá el pedido en su panel con detalles de productos, dirección y método de pago.
* RF6.03. El repartidor podrá actualizar el estado de la entrega: “en camino”, “entregado” o “cancelado”.
* RF6.04. En caso de cancelación, deberá registrar el motivo.
* RF6.05. El sistema guardará fecha de asignación, hora de salida y hora de entrega.

**RF7. Seguimiento de pedidos (tracking)**

El sistema permitirá al cliente monitorear sus pedidos en tiempo real.

* RF7.01. El cliente visualizará el estado: pendiente, confirmado, en preparación, listo para enviar, en camino, entregado o cancelado.
* RF7.02. Cada cambio de estado quedará registrado en el historial del pedido con fecha y hora.
* RF7.03. El sistema enviará notificaciones al cliente por correo o mensaje.

**RF8. Funcionalidades complementarias**

* RF8.01. Gestión de promociones y descuentos especiales.
* RF8.02. Gestión de testimonios de clientes (previa aprobación por el administrador).
* RF8.03. Gestión de artículos en el blog (crear, modificar, publicar).
* RF8.04. Formulario de contacto para consultas de clientes.

**RF9. Reportes y estadísticas**

El sistema generará reportes automáticos para apoyo en la toma de decisiones.

* RF9.01. Reporte de ventas diarias, semanales y mensuales.
* RF9.02. Reporte de productos más vendidos y menos vendidos.
* RF9.03. Reporte de clientes frecuentes y montos de compra.
* RF9.04. Reporte de desempeño de repartidores (entregas completadas vs canceladas).

**Requisitos No Funcionales (RNF)**

**RNF1. Usabilidad**

* RNF1.01. La interfaz será intuitiva, fácil de navegar y responsive (compatible con móviles, tablets y PC).
* RNF1.02. Los formularios incluirán validaciones inmediatas de campos obligatorios.
* RNF1.03. El sistema utilizará íconos, colores y mensajes claros para guiar al usuario.

**RNF2. Rendimiento y capacidad**

* RNF2.01. El tiempo de respuesta de cualquier operación no excederá los 3 segundos bajo carga normal.
* RNF2.02. El sistema soportará al menos 100 usuarios concurrentes.
* RNF2.03. La base de datos deberá manejar al menos 100,000 registros de pedidos sin afectar el rendimiento.

**RNF3. Seguridad**

* RNF3.01. Las contraseñas se encriptarán con bcrypt.
* RNF3.02. Se implementará control de acceso por roles (Spring Security).
* RNF3.03. El sistema tendrá protección contra ataques comunes: SQL Injection, XSS y CSRF.
* RNF3.04. Se establecerá política de sesión segura con expiración por inactividad.

**RNF4. Confiabilidad y disponibilidad**

* RNF4.01. Disponibilidad mínima del sistema: 95% mensual.
* RNF4.02. Se realizarán respaldos automáticos diarios de la base de datos.
* RNF4.03. Se contará con mecanismos de recuperación ante fallos (rollback de transacciones).

**RNF5. Escalabilidad y mantenibilidad**

* RNF5.01. El sistema permitirá integrar nuevas sucursales y zonas de reparto sin reestructurar la base de datos.
* RNF5.02. El código se desarrollará bajo arquitectura MVC y será modular para facilitar mantenimiento.
* RNF5.03. La base de datos estará normalizada hasta 3FN para evitar redundancia y anomalías.

**RNF6. Interoperabilidad**

* RNF6.01. Integración vía API con pasarelas de pago (Yape, Plin, tarjetas, transferencias).
* RNF6.02. Integración con SUNAT para validación de comprobantes electrónicos.
* RNF6.03. Exportación de reportes en formatos estándar (CSV, XLSX, PDF).

### **Modelo lógico de la Base de Datos**

#### **Descripción de las entidades identificadas**

Entidad: provincia

* Descripción: Catálogo de provincias donde el restaurante puede operar.
* Atributos:
  + id\_provincia (PK)
  + nombre
  + igv\_aplica
* Relaciones: Una provincia puede tener varias zonas de reparto (*1:N con zonas\_reparto*).

Entidad: zonas\_reparto

* Descripción: Define las zonas específicas de cobertura de delivery.
* Atributos:
  + id\_zonas\_reparto (PK)
  + provincia\_id (FK -> provincia)
  + codigo\_zona, cobertura, activo
* Relaciones: Una zona de reparto pertenece a una provincia (*N:1*) y puede tener varias sucursales (*1:N*).

Entidad: sucursal

* Descripción: Representa las sedes físicas del restaurante.
* Atributos:
  + id\_sucursal (PK)
  + zona\_reparto\_id (FK -> zonas\_reparto)
  + codigo\_sucursal, direccion, telefono, horario\_atencion, latitud, longitud, estado
* Relaciones: Una sucursal atiende pedidos y tiene usuarios internos asociados (*1:N con usuario y pedido*).

Entidad: persona

* Descripción: Datos personales básicos de clientes y empleados.
* Atributos:
  + id\_persona (PK)
  + nombre, apellido, tipo\_documento, numero\_documento, telefono, correo, fecha\_registro
* Relaciones: Una persona puede ser cliente o empleado mediante la entidad usuario.

Entidad: usuario

* Descripción: Representa la cuenta del sistema de cada persona.
* Atributos:
  + persona\_id (PK, FK -> persona)
  + sucursal\_id (FK -> sucursal)
  + rol, correo, contrasena, estado, cambio\_contrasena
* Relaciones: Un usuario puede ser administrador, recepcionista, repartidor o cliente.

Entidad: categoria

* Descripción: Agrupa productos según su tipo.
* Atributos:
  + id\_categoria (PK)
  + nombre, estado
* Relaciones: Una categoría puede tener muchos productos (*1:N con producto*).

Entidad: producto

* Descripción: Representa los platos saludables disponibles para la venta.
* Atributos:
  + id\_producto (PK)
  + categoria\_id (FK -> categoria)
  + nombre, descripcion, precio\_unitario, descuento, disponible, imagen\_url
* Relaciones: Se relaciona con detalle\_pedido.

Entidad: pedido

* Descripción: Cabecera de las órdenes generadas por los clientes.
* Atributos:
  + id\_pedido (PK)
  + sucursal\_id (FK -> sucursal)
  + cliente\_id (FK -> persona)
  + direccion\_entrega, subtotal, impuesto, monto\_total, total\_pagado, estado\_pago\_global, fecha\_pedido, fecha\_entrega
* Relaciones: Se relaciona con detalle\_pedido, pago, comprobante, entrega e historial\_estado\_pedido.

Entidad: detalle\_pedido

* Descripción: Representa los ítems del pedido.
* Atributos:
  + id\_detalle\_pedido (PK)
  + pedido\_id (FK -> pedido)
  + producto\_id (FK -> producto)
  + cantidad, precio\_unitario, descuento\_item, subtotal\_item

Entidad: metodos\_pago

* Descripción: Catálogo de formas de pago admitidas.
* Atributos:
  + id\_metodos\_pago (PK)
  + nombre, activo

Entidad: pago

* Descripción: Registra cada transacción parcial o total asociada a un pedido.
* Atributos:
  + id\_pago (PK)
  + pedido\_id (FK -> pedido)
  + metodo\_pago\_id (FK -> metodos\_pago)
  + monto\_pagado, fecha\_pago, moneda, referencia\_pago

Entidad: comprobante

* Descripción: Documento tributario asociado al pedido.
* Atributos:
  + pedido\_id (PK, FK -> pedido)
  + tipo\_comprobante, serie, numero\_comprobante, fecha\_emision, moneda, subtotal, igv, total, estado, archivo\_url, observaciones

Entidad: historial\_estado\_pedido

* Descripción: Registro de cambios de estado de un pedido.
* Atributos:
  + id\_estado\_pedido (PK)
  + pedido\_id (FK -> pedido)
  + estado, fecha\_estado

Entidad: entrega

* Descripción: Controla las asignaciones de pedidos a repartidores.
* Atributos:
  + id\_entrega (PK)
  + pedido\_id (FK -> pedido)
  + trabajador\_id (FK -> persona)
  + repartidor\_id (FK -> persona)
  + fecha\_asignacion, hora\_salida, hora\_entrega, estado\_entrega, observaciones

Entidad: contacto

* Descripción: Registra los mensajes enviados por clientes.
* Atributos:
  + id\_contacto (PK)
  + nombre, correo, telefono, mensaje, fecha\_envio

Entidades JSON dinámicas

* promociones\_json, blog\_articulos\_json, testimonios\_json
* Atributos:
  + id (PK)
  + data JSON

#### **Esquema del modelo lógico**

|  |  |
| --- | --- |
| RELACIÓN | ATRIBUTOS |
| provincia | **id\_provincia (PK),** nombre, igv\_aplica |
| zonas\_reparto | **id\_zonas\_reparto (PK),** provincia\_id (FK), codigo\_zona, cobertura, activo |
| sucursal | **id\_sucursal (PK),** zona\_reparto\_id (FK), codigo\_sucursal, direccion, telefono, horario\_atencion, latitud, longitud, estado |
| persona | **id\_persona (PK),** nombre, apellido, tipo\_documento, numero\_documento, telefono, correo, fecha\_registro |
| usuario | **persona\_id (PK, FK),** sucursal\_id (FK), rol, correo, contrasena, estado, cambio\_contrasena |
| categoria | id\_categoria (PK), nombre, estado |
| producto | **id\_producto (PK),** categoria\_id (FK), nombre, descripcion, precio\_unitario, descuento, disponible, imagen\_url |
| pedido | **id\_pedido (PK),** sucursal\_id (FK), cliente\_id (FK), direccion\_entrega, subtotal, impuesto, monto\_total, total\_pagado, estado\_pago\_global, fecha\_pedido, fecha\_entrega |
| detalle\_pedido | **id\_detalle\_pedido (PK),** pedido\_id (FK), producto\_id (FK), cantidad, precio\_unitario, descuento\_item, subtotal\_item |
| metodos\_pago | id\_metodos\_pago (PK), nombre, activo |
| pago | **id\_pago (PK),** pedido\_id (FK), metodo\_pago\_id (FK), monto\_pagado, fecha\_pago, moneda, referencia\_pago |
| comprobante | **pedido\_id (PK, FK),** tipo\_comprobante, serie, numero\_comprobante, fecha\_emision, moneda, subtotal, igv, total, estado, archivo\_url, observaciones |
| historial\_estado\_pedido | **id\_estado\_pedido (PK),** pedido\_id (FK), estado, fecha\_estado |
| entrega | **id\_entrega (PK),** pedido\_id (FK), trabajador\_id (FK), repartidor\_id (FK), fecha\_asignacion, hora\_salida, hora\_entrega, estado\_entrega, observaciones |
| contacto | **id\_contacto (PK),** nombre, correo, telefono, mensaje, fecha\_envio |
| promociones\_json | **id (PK),** data JSON |
| blog\_articulos\_json | **id (PK),** data JSON |
| testimonios\_json | **id (PK),** data JSON |

#### **Diseño lógico de la Base de Datos**

El modelo de base de datos diseñado para el sistema *Sabores Los Huayachos* cumple con los principios de normalización hasta la Tercera Forma Normal (3FN), asegurando consistencia, ausencia de redundancia y eliminación de anomalías de inserción, actualización y eliminación.

**Primera Forma Normal (1FN)**

* Todas las tablas tienen atributos atómicos (sin valores repetitivos ni multivaluados).
* Ejemplo: en la tabla producto, el atributo nombre almacena un único valor por fila; el stock se maneja como un número entero, no como una lista de lotes.

**Segunda Forma Normal (2FN)**

* Cada tabla con clave primaria compuesta se descompuso en relaciones adicionales para eliminar dependencias parciales.
* Ejemplo: detalle\_pedido depende de la clave compuesta (pedido\_id, producto\_id). Atributos como cantidad, precio\_unitario, subtotal\_item dependen de toda la clave, no solo de una parte.

**Tercera Forma Normal (3FN)**

* No existen dependencias transitivas (atributos que dependan de otros atributos no clave).
* Ejemplo: en pedido, atributos como subtotal, impuesto y monto\_total dependen únicamente de id\_pedido. La información de cliente se gestiona en la tabla persona, evitando duplicación en pedido.

**Verificación en el modelo existente**

* provincia: cumple 3FN, no hay dependencias transitivas.
* zonas\_reparto: cada zona depende de una provincia, relación clara 1:N.
* sucursal: depende de zona y almacena datos atómicos (dirección, teléfono).
* persona / usuario: separación clara entre datos personales y credenciales de acceso -> elimina redundancia.
* categoria / producto: productos dependen de categorías, atributos simples.
* pedido: no almacena detalles de productos directamente -> se delega en detalle\_pedido.
* detalle\_pedido: normalizado en 2FN y 3FN, sin duplicidad de datos.
* metodos\_pago / pago: un pago está ligado a un pedido y un método, evitando valores repetidos.
* comprobante: ligado directamente a un pedido -> 1:1, sin redundancias.
* historial\_estado\_pedido: registra cambios en estados, evitando tener múltiples columnas de estado en pedido.
* entrega: separa la asignación de repartidores, normalizado correctamente.
* contacto: almacena mensajes únicos por cliente.
* tablas JSON (promociones, blog, testimonios): permiten flexibilidad para datos dinámicos, pero no afectan la normalización de las relaciones principales.

**Conclusión del Diseño Lógico**

* Todas las entidades cumplen con 1FN, 2FN y 3FN.
* Se evita duplicidad de información gracias a la separación de tablas (detalle\_pedido, historial\_estado\_pedido, pago).
* El modelo está preparado para garantizar integridad referencial mediante claves primarias y foráneas.
* El uso de tablas puente (detalle\_pedido) asegura que la relación N:M entre pedidos y productos esté correctamente normalizada.
* Tablas auxiliares (metodos\_pago, categoria, zonas\_reparto) permiten consistencia y reutilización de catálogos.

## Modelo Físico de la Base de Datos

### **Indicar el SGBD, el fabricante, la versión y distribución**

* Sistema Gestor de Base de Datos (SGBD): MySQL
* Fabricante: Oracle Corporation
* Versión recomendada: MySQL 8.0.36 (o superior estable de la rama 8.x)
* Distribución:
  + Compatible con Windows, Linux y macOS.
  + Se puede implementar de manera local, en servidores on-premise o en la nube (ejemplo: MySQL Community Edition, Amazon RDS for MySQL, Azure Database for MySQL).
  + Motor de almacenamiento recomendado: InnoDB (soporta transacciones ACID, claves foráneas e integridad referencial).

**Justificación de la elección**

1. Compatibilidad: MySQL es ampliamente soportado por frameworks como Spring Boot, JPA/Hibernate, lo cual facilita la integración con el backend del sistema.
2. Rendimiento: Ofrece buena velocidad en entornos de alta concurrencia, adecuado para un sistema de ventas en línea con múltiples usuarios.
3. Escalabilidad: Permite crecer a múltiples sucursales y zonas de reparto, manteniendo consistencia de datos.
4. Soporte para JSON: Incluye tipos de datos JSON nativos, utilizados en tablas dinámicas como promociones, blogs y testimonios.
5. Costo: La Community Edition es gratuita y de código abierto, con amplia comunidad de soporte.

### **Detalle de cada tabla**

**Tabla: CLIENTE**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de columna | Descripción / restricciones | Tipo de datos | Permite NULL |
| id\_cliente (PK) | Identificador único del cliente | INT AUTO\_INCREMENT | NO |
| nombre | Nombres del cliente | VARCHAR(80) | NO |
| apellido | Apellidos del cliente | VARCHAR(80) | NO |
| correo | Correo electrónico | VARCHAR(100) | NO (Único) |
| telefono | Número de contacto | VARCHAR(20) | SI |
| fecha\_registro | **Fecha de alta del cliente** | **DATE** | **NO** |

**Índices**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de índice | Descripción | Primario | Único |
| cliente\_PK | Índice de clave primaria | SI | SI |
| cliente\_UQ | Índice único en correo electrónico | NO | SI |

**Tabla: CATEGORIA**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de columna | Descripción | Tipo de datos | Permite NULL |
| id\_categoria (PK) | Identificador único de categoría | INT | NO |
| nombre | Nombre de la categoría | VARCHAR(100) | NO |
| estado | Estado (1=Activa, 0=Inactiva) | TINYINT | NO |

**Índices**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de índice | Descripción | Primario | Único |
| categoria\_PK | Clave primaria | SI | SI |

**Tabla: PRODUCTO**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de columna | Descripción / restricciones | Tipo de datos | Permite NULL |
| id\_producto (PK) | Identificador único del producto | INT AUTO\_INCREMENT |  |
| id\_categoria (FK) | Categoría a la que pertenece | INT |  |
| nombre | Nombre del producto | VARCHAR(100) |  |
| descripcion | Descripción detallada | VARCHAR(255) | SI |
| precio\_unitario | Precio base | DECIMAL(10,2) |  |
| descuento | Descuento aplicado | DECIMAL(10,2) | SI |
| disponible | Disponibilidad (1=Sí, 0=No) | TINYINT |  |
| imagen\_url | URL de la imagen | VARCHAR(200) | SI |

**Índices**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de índice | Descripción | Primario | Único |
| producto\_PK | Índice de clave primaria | SI | SI |
| producto\_categoria | Índice de búsqueda por categoría | NO | NO |

**Tabla: PEDIDO**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de columna | Descripción / restricciones | Tipo de datos | Permite NULL |
| id\_pedido (PK) | Identificador único del pedido | INT AUTO\_INCREMENT | NO |
| cliente\_id (FK) | Cliente que realizó el pedido | INT | NO |
| sucursal\_id (FK) | Sucursal donde se generó el pedido | INT | NO |
| direccion\_entrega | Dirección proporcionada por el cliente | VARCHAR(150) | NO |
| subtotal | Subtotal del pedido | DECIMAL(10,2) | NO |
| impuesto | Impuesto (IGV) aplicado | DECIMAL(10,2) | NO |
| monto\_total | Total a pagar | DECIMAL(10,2) | NO |
| total\_pagado | Monto efectivamente pagado | DECIMAL(10,2) | SI |
| estado\_pago\_global | Estado del pago (Pendiente, Parcial, Completo) | VARCHAR(20) | NO |
| fecha\_pedido | Fecha y hora del pedido | DATETIME | NO |
| fecha\_entrega | Fecha programada de entrega | DATE | SI |

**Índices**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de índice | Descripción | Primario | Único |
| pedido\_PK | Índice de clave primaria | SI | SI |
| pedido\_cliente | Índice de búsqueda por cliente | NO | NO |
| pedido\_fecha | Índice para consultas por fecha | NO | NO |

**Tabla: DETALLE\_PEDIDO**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de columna | Descripción / restricciones | Tipo de datos | Permite NULL |
| id\_detalle (PK) | Identificador único del detalle | INT AUTO\_INCREMENT | NO |
| id\_pedido (FK) | Pedido al que pertenece | INT | NO |
| id\_producto (FK) | Producto incluido en el pedido | INT | NO |
| cantidad | Cantidad solicitada | INT | NO |
| precio\_unitario | Precio del producto en ese momento | DECIMAL(10,2) | NO |
| descuento\_item | Descuento aplicado al ítem | DECIMAL(10,2) | SI |
| subtotal\_item | Subtotal del ítem | DECIMAL(10,2) | NO |

**Índices**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de índice | Descripción | Primario | Único |
| detalle\_PK | Índice de clave primaria | SI | SI |
| detalle\_pedido\_FK | Índice de búsqueda por pedido | NO | NO |

**Tabla: PROVINCIA**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de columna | Descripción / restricciones | Tipo de datos | Permite NULL |
| id\_provincia (PK) | Identificador único de la provincia | INT | NO |
| nombre | Nombre de la provincia | VARCHAR(50) | NO |
| igv\_aplica | Indica si aplica IGV (0=No, 1=Sí) | TINYINT | NO |

**Índices**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de índice | Descripción | Primario | Único |
| provincia\_PK | Clave primaria | SI | SI |

**Tabla: ZONAS\_REPARTO**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de columna | Descripción | Tipo de datos | Permite NULL |
| id\_zonas\_reparto (PK) | Identificador único de la zona de reparto | INT | NO |
| provincia\_id (FK) | Provincia asociada | INT | NO |
| codigo\_zona | Código de la zona | VARCHAR(20) | NO |
| cobertura | Nombre / descripción de cobertura | VARCHAR(100) | NO |
| activo | Estado (1=Activo, 0=Inactivo) | TINYINT | NO |

**Índices**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de índice | Descripción | Primario | Único |
| zonas\_PK | Clave primaria | SI | SI |
| zonas\_provincia\_FK | Índice para búsqueda por provincia | NO | NO |

**Tabla: SUCURSAL**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de columna | Descripción | Tipo de datos | Permite NULL |
| id\_sucursal (PK) | Identificador único de la sucursal | INT | NO |
| zona\_reparto\_id (FK) | Zona de reparto asociada | INT | NO |
| codigo\_sucursal | Código único de sucursal | VARCHAR(20) | NO (Único) |
| direccion | Dirección de la sucursal | VARCHAR(255) | NO |
| telefono | Teléfono de contacto | VARCHAR(100) | SI |
| horario\_atencion | Horario de atención | VARCHAR(100) | SI |
| latitud | Latitud (coordenadas) | DECIMAL(11,8) | SI |
| longitud | Longitud (coordenadas) | DECIMAL(11,8) | SI |
| estado | Estado de la sucursal (1=Activa, 0=Inactiva) | TINYINT | NO |

**Índices**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de índice | Descripción | Primario | Único |
| sucursal\_PK | Clave primaria | SI | SI |
| sucursal\_codigo\_UQ | Código de sucursal único | NO | NO |

**Tabla: PERSONA**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de columna | Descripción | Tipo de datos | Permite NULL |
| id\_persona (PK) | Identificador único de persona | INT | NO |
| nombre | Nombre | VARCHAR(80) | NO |
| apellido | Apellido | VARCHAR(80) | NO |
| tipo\_documento | Tipo de documento (DNI, CE, etc.) | VARCHAR(10) | NO |
| numero\_documento | Número de documento único | VARCHAR(45) | NO (Único) |
| telefono | Teléfono de la persona | VARCHAR(30) | SI |
| correo | Correo electrónico | VARCHAR(255) | SI |
| fecha\_registro | Fecha de registro | DATE | NO |

**Índices**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de índice | Descripción | Primario | Único |
| persona\_PK | Clave primaria | SI | SI |
| persona\_documento\_UQ | Documento único | NO | SI |

**Tabla: USUARIO**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de columna | Descripción | Tipo de datos | Permite NULL |
| persona\_id (PK, FK) | Identificador de persona | INT | NO |
| sucursal\_id (FK) | Sucursal a la que pertenece | INT | NO |
| rol | Rol del usuario (ADMIN, RECEPCION, etc.) | ENUM | NO |
| correo | Correo de acceso (único) | VARCHAR(255) | NO |
| contrasena | Contraseña encriptada | VARCHAR(255) | NO |
| estado | Estado de la cuenta | TINYINT | NO |
| cambio\_contrasena | Indicador de cambio de contraseña | TINYINT | NO |

**Índices**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de índice | Descripción | Primario | Único |
| usuario\_PK | Clave primaria | SI | SI |
| usuario\_correo\_UQ | Correo único | NO | SI |

**Tabla: METODOS\_PAGO**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de columna | Descripción | Tipo de datos | Permite NULL |
| id\_metodos\_pago (PK) | Identificador único del método de pago | INT | NO |
| nombre | Nombre del método (Yape, Plin, etc.) | VARCHAR(100) | NO |
| activo | Estado (1=Activo, 0=Inactivo) | TINYINT | NO |

**Índices**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de índice | Descripción | Primario | Único |
| metodos\_pago\_PK | Clave primaria | SI | SI |

**Tabla: PAGO**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de columna | Descripción | Tipo de datos | Permite NULL |
| id\_pago (PK) | Identificador del pago | INT | NO |
| pedido\_id (FK) | Pedido asociado | INT | NO |
| metodo\_pago\_id(FK) | Método de pago utilizado | INT | NO |
| monto\_pagado | Monto del pago realizado | DECIMAL(10,2) | NO |
| fecha\_pago | Fecha del pago | DATE | NO |
| moneda | Moneda utilizada (PEN/USD) | VARCHAR(3) | NO |
| referencia\_pago | Código/Referencia de la transacción | VARCHAR(100) | SI |

**Índices**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de índice | Descripción | Primario | Único |
| pago\_PK | Clave primaria | SI | SI |
| pago\_pedido\_FK | Índice por pedido | NO | NO |

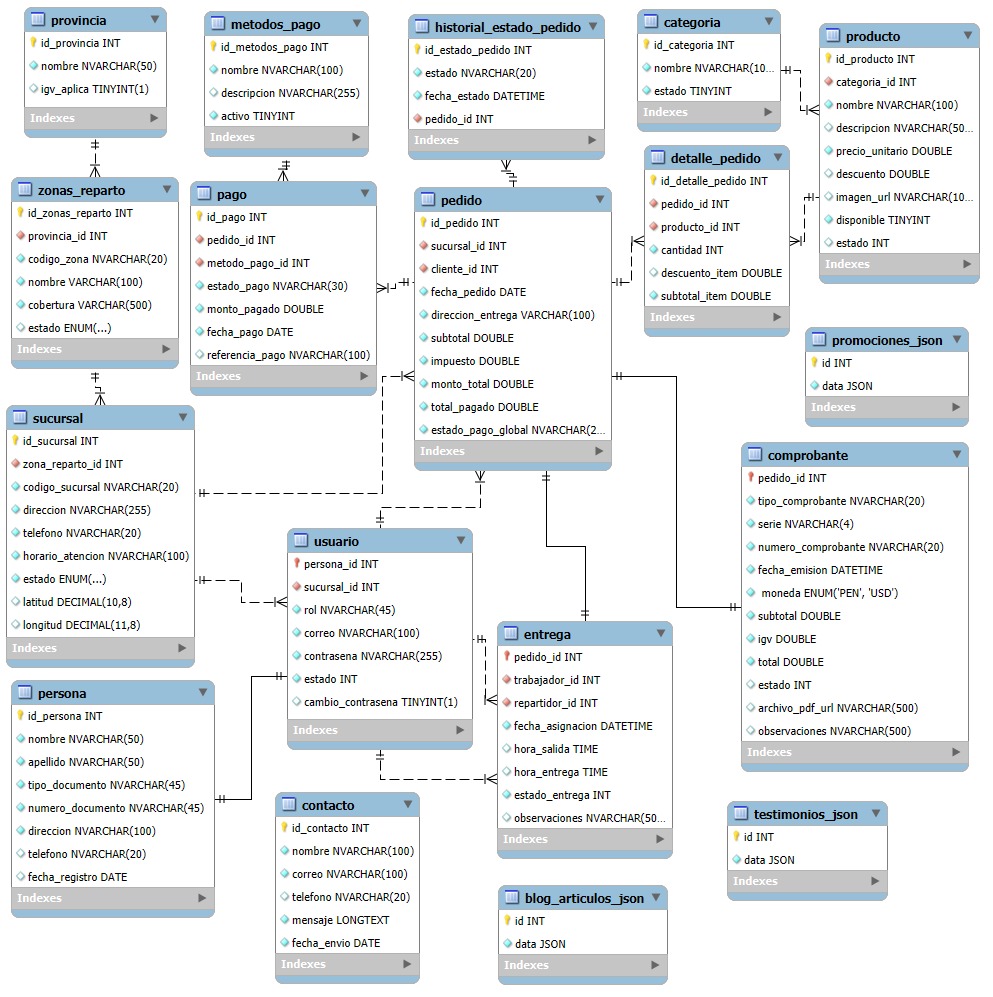
**Tabla: COMPROBANTE**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de columna | Descripción | Tipo de datos | Permite NULL |
| pedido\_id (PK, FK) | Pedido al que corresponde | INT | NO |
| tipo\_comprobante | Tipo de comprobante (Boleta o Factura) | VARCHAR(20) | NO |
| serie | Serie del comprobante | VARCHAR(4) | NO |
| numero\_comprobante | Número correlativo | VARCHAR(20) | NO |
| fecha\_emision | Fecha de emisión | DATETIME | NO |
| moneda | Moneda del comprobante | ENUM('PEN','USD') | NO |
| subtotal | Subtotal del pedido | DECIMAL(10,2) | NO |
| igv | Impuesto aplicado | DECIMAL(10,2) | NO |
| total | Total del comprobante | DECIMAL(10,2) | NO |
| estado | Estado del comprobante | INT | NO |
| archivo\_url | Ruta al archivo generado (PDF/XML) | VARCHAR(500) | SI |
| observaciones | Notas adicionales | VARCHAR(500) | SI |

**Índices**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre de índice | Descripción | Primario | Único |
| comprobante\_PK | Clave primaria | SI | SI |

### **Diagrama físico**

****

### **Diccionario de datos**

**Tabla: PROVINCIA**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| id\_provincia | INT | — | PK | No | Identificador único de la provincia |
| nombre | VARCHAR | 50 | — | No | Nombre de la provincia |
| igv\_aplica | TINYINT | 1 | — | No | Indica si aplica IGV (0=No, 1=Sí) |

**Tabla: ZONAS\_REPARTO**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| id\_zonas\_reparto | INT | — | PK | No | Identificador único de la zona de reparto |
| provincia\_id | INT | — | FK | No | Relación con la provincia |
| codigo\_zona | VARCHAR | 20 | — | No | Código asignado a la zona |
| cobertura | VARCHAR | 100 | — | No | Nombre / descripción de la zona |
| activo | TINYINT | 1 | — | No | Estado (1=Activo, 0=Inactivo) |

**Tabla: SUCURSAL**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| id\_sucursal | INT | — | PK | No | Identificador único de la sucursal |
| zona\_reparto\_id | INT | — | FK | No | Zona de reparto de la sucursal |
| codigo\_sucursal | VARCHAR | 20 | — | No | Código único de sucursal |
| direccion | VARCHAR | 255 | — | No | Dirección de la sucursal |
| telefono | VARCHAR | 100 | — | Sí | Teléfono de contacto |
| horario\_atencion | VARCHAR | 100 | — | Sí | Horarios de atención |
| latitud | DECIMAL | 11,8 | — | Sí | Latitud de ubicación |
| longitud | DECIMAL | 11,8 | — | Sí | Longitud de ubicación |
| estado | TINYINT | 1 | — | No | Estado (1=Activa, 0=Inactiva) |

**Tabla: PERSONA**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| id\_persona | INT | — | PK | No | Identificador único de persona |
| nombre | VARCHAR | 80 | — | No | Nombre de la persona |
| apellido | VARCHAR | 80 | — | No | Apellido de la persona |
| tipo\_documento | VARCHAR | 10 | — | No | Tipo de documento (DNI, CE, Pasaporte) |
| numero\_documento | VARCHAR | 45 | — | No | Número de documento único |
| telefono | VARCHAR | 30 | — | Sí | Número telefónico |
| correo | VARCHAR | 255 | — | Sí | Correo electrónico |
| fecha\_registro | DATE | — | — | No | Fecha de registro en el sistema |

**Tabla: USUARIO**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| persona\_id | INT | — | PK/FK | No | Identificador de persona asociada |
| sucursal\_id | INT | — | FK | No | Sucursal donde labora el usuario |
| rol | ENUM | — | — | No | Rol (ADMIN, RECEPCION, REPARTIDOR, CLIENTE) |
| correo | VARCHAR | 255 | — | No | Correo de acceso |
| contrasena | VARCHAR | 255 | — | No | Contraseña encriptada |
| estado | TINYINT | 1 | — | No | Estado (1=Activo, 0=Inactivo) |
| cambio\_contrasena | TINYINT | 1 | — | No | Indicador de cambio de contraseña |

**Tabla: CATEGORIA**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| id\_categoria | INT | — | PK | No | Identificador único |
| nombre | VARCHAR | 100 | — | No | Nombre de la categoría |
| estado | TINYINT | 1 | — | No | Estado (1=Activo, 0=Inactivo) |

**Tabla: PRODUCTO**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| id\_producto | INT | — | PK | No | Identificador único |
| categoria\_id | INT | — | FK | No | Categoría a la que pertenece |
| nombre | VARCHAR | 100 | — | No | Nombre del producto |
| descripcion | **VARCHAR** | **255** | **—** | **Sí** | **Descripción detallada** |
| precio\_unitario | **DECIMAL** | **10,2** | **—** | **No** | **Precio unitario** |
| descuento | **DECIMAL** | **10,2** | **—** | **Sí** | **Descuento aplicado** |
| disponible | **TINYINT** | **1** | **—** | **No** | **Disponibilidad (1=Sí, 0=No)** |
| imagen\_url | **VARCHAR** | **200** | **—** | **Sí** | **Ruta o URL de la imagen del producto** |

**Tabla: PEDIDO**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| id\_pedido | INT | — | PK | No | Identificador único |
| sucursal\_id | INT | — | FK | No | Sucursal que gestiona el pedido |
| cliente\_id | INT | — | FK | No | Cliente que realiza el pedido |
| direccion\_entrega | VARCHAR | 150 | — | No | Dirección de entrega |
| subtotal | DECIMAL | 10,2 | — | No | Subtotal calculado |
| impuesto | DECIMAL | 10,2 | — | No | Impuesto aplicado (IGV) |
| monto\_total | DECIMAL | 10,2 | — | No | Total general del pedido |
| total\_pagado | DECIMAL | 10,2 | — | Sí | Total efectivamente pagado |
| estado\_pago\_global | VARCHAR | 20 | — | No | Estado del pago (Pendiente, Parcial, Pagado) |
| fecha\_pedido | DATETIME | — | — | No | Fecha y hora de creación |
| fecha\_entrega | DATE | — | — | Sí | Fecha de entrega |

**Tabla: DETALLE\_PEDIDO**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| id\_detalle\_pedido | INT | — | PK | No | Identificador del detalle |
| pedido\_id | INT | — | FK | No | Pedido asociado |
| producto\_id | INT | — | FK | No | Producto solicitado |
| cantidad | INT | — | — | No | Cantidad solicitada |
| precio\_unitario | DECIMAL | 10,2 | — | No | Precio del producto al momento |
| descuento\_item | DECIMAL | 10,2 | — | Sí | Descuento aplicado |
| subtotal\_item | DECIMAL | 10,2 | — | No | Subtotal del ítem |

**Tabla: METODOS\_PAGO**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| id\_metodos\_pago | INT | — | PK | No | Identificador único del método |
| nombre | VARCHAR | 100 | — | No | Nombre (Yape, Plin, Tarjeta, etc.) |
| activo | TINYINT | 1 | — | No | Estado (1=Activo, 0=Inactivo) |

**Tabla: PAGO**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| id\_pago | INT | — | PK | No | Identificador único del pago |
| pedido\_id | INT | — | FK | No | Pedido asociado |
| metodo\_pago\_id | INT | — | FK | No | Método de pago utilizado |
| monto\_pagado | DECIMAL | 10,2 | — | No | Monto del pago |
| fecha\_pago | DATE | — | — | No | Fecha en que se realizó el pago |
| moneda | VARCHAR | 3 | — | No | Moneda (PEN, USD) |
| referencia\_pago | VARCHAR | 100 | — | Sí | Referencia o código de transacción (voucher) |

**Tabla: COMPROBANTE**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| pedido\_id | INT | — | PK/FK | No | Pedido asociado (1:1) |
| tipo\_comprobante | VARCHAR | 20 | — | No | Tipo (Boleta o Factura) |
| serie | VARCHAR | 4 | — | No | Serie del comprobante |
| numero\_comprobante | VARCHAR | 20 | — | No | Número correlativo del comprobante |
| fecha\_emision | DATETIME | — | — | No | Fecha y hora de emisión |
| moneda | ENUM | — | — | No | Moneda utilizada (PEN, USD) |
| subtotal | DECIMAL | 10,2 | — | No | Subtotal antes de impuestos |
| igv | DECIMAL | 10,2 | — | No | IGV aplicado |
| total | DECIMAL | 10,2 | — | No | Total a pagar |
| estado | INT | — | — | No | Estado del comprobante |
| archivo\_url | VARCHAR | 500 | — | Sí | Ruta del archivo PDF/XML generado |
| observaciones | VARCHAR | 500 | — | Sí | Notas adicionales |

**Tabla: HISTORIAL\_ESTADO\_PEDIDO**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| id\_estado\_pedido | INT | — | PK | No | Identificador del registro |
| pedido\_id | INT | — | FK | No | Pedido al que pertenece |
| estado | VARCHAR | 30 | — | No | Estado (Pendiente, Confirmado, etc.) |
| fecha\_estado | DATETIME | — | — | No | Fecha y hora del cambio de estado |

**Tabla: ENTREGA**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| id\_entrega | INT | — | PK | No | Identificador único de la entrega |
| pedido\_id | INT | — | FK | No | Pedido asociado |
| trabajador\_id | INT | — | FK | Sí | Trabajador que asigna la entrega |
| repartidor\_id | INT | — | FK | No | Repartidor encargado de la entrega |
| fecha\_asignacion | DATETIME | — | — | No | Fecha de asignación |
| hora\_salida | TIME | — | — | Sí | Hora de salida del pedido |
| hora\_entrega | TIME | — | — | Sí | Hora en que se realizó la entrega |
| estado\_entrega | INT | — | — | No | Estado de la entrega (Asignado, En camino…) |
| observaciones | VARCHAR | 500 | — | Sí | Comentarios u observaciones adicionales |

**Tabla: CONTACTO**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| id\_contacto | INT | — | PK | No | Identificador único del contacto |
| nombre | VARCHAR | 100 | — | No | Nombre de la persona que consulta |
| correo | VARCHAR | 100 | — | Sí | Correo electrónico |
| telefono | VARCHAR | 30 | — | Sí | Teléfono de contacto |
| mensaje | LONGTEXT | — | — | No | Mensaje enviado |
| fecha\_envio | DATE | — | — | No | Fecha en que se registró la consulta |

**Tabla: PROMOCIONES\_JSON**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| id | INT | — | PK | No | Identificador único |
| data | JSON | — | — | No | Contenido dinámico en formato JSON |

**Tabla: BLOG\_ARTICULOS\_JSON**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| id | INT | — | PK | No | Identificador único |
| data | JSON | — | — | No | Contenido de artículos en formato JSON |

**Tabla: TESTIMONIOS\_JSON**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Campo | Tipo de dato | Tamaño | PK/FK | Nulo | Descripción |
| id | INT | — | PK | No | Identificador único |
| data | JSON | — | — | No | Testimonios de clientes en formato JSON |

### **Scripts del proceso de construcción de la BBDD**

Se adjunta al informe como archivo en formato SQL con nombre **BDSabhuay.sql**

### **Copias de respaldo (pendiente)**

Las copias de respaldo son un componente esencial de la administración de la base de datos, ya que aseguran la recuperación de la información en caso de fallas técnicas, pérdidas accidentales, ataques de seguridad o desastres.

**Estrategia de Respaldo Propuesta**

1. Tipo de respaldos

* Respaldo completo (Full Backup):

Se realizará una copia completa de toda la base de datos una vez al día, en horario nocturno, cuando la carga de trabajo del sistema es mínima.

* Respaldo incremental:

Cada 4 horas se realizará una copia incremental basada en los binlogs de MySQL, para registrar únicamente los cambios desde el último respaldo completo.

* Respaldo de emergencia:

Se podrá ejecutar bajo demanda previo a actualizaciones críticas del sistema (ej. cambios de estructura o migraciones).

2. Herramientas utilizadas

* mysqldump: Para respaldos lógicos en formato .sql.
* mysqlpump: Para respaldos paralelos y más rápidos en bases de datos grandes.
* Percona XtraBackup (opcional): Para respaldos físicos sin bloquear la base de datos.

3. Almacenamiento de respaldos

* Copias almacenadas localmente en el servidor en una carpeta protegida /backups/huayachos/.
* Copias duplicadas en un servicio de nube seguro (ejemplo: Google Drive, AWS S3 o Azure Storage).
* Políticas de retención:
  + Respaldos completos: retención de 30 días.
  + Respaldos incrementales: retención de 7 días.

4. Procedimiento de restauración

* Validar la copia más reciente.
* Restaurar con mysql < backup.sql en caso de respaldo lógico.
* Restaurar con xtrabackup --copy-back en caso de respaldo físico.
* Verificación post-restauración: prueba de integridad ejecutando consultas de verificación en tablas críticas (pedido, pago, producto).

5. Frecuencia y responsables

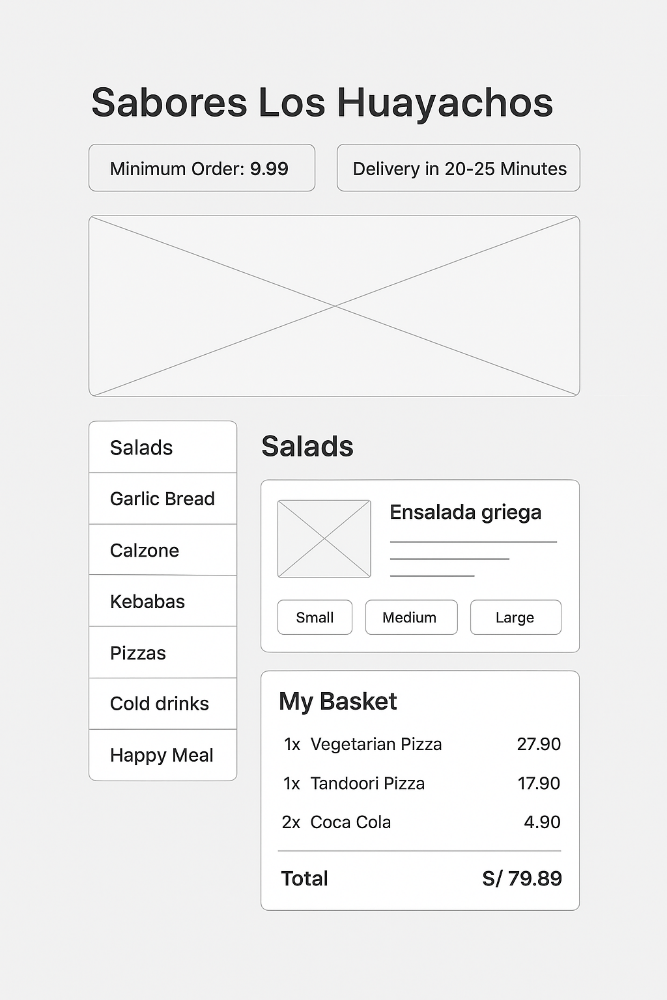
* Frecuencia:
  + Respaldo completo diario.
  + Incrementales cada 4 horas.
* Responsable: Administrador de Base de Datos (DBA) designado.
* Supervisión: El sistema notificará automáticamente al administrador mediante correo/log cuando el respaldo se complete o falle.

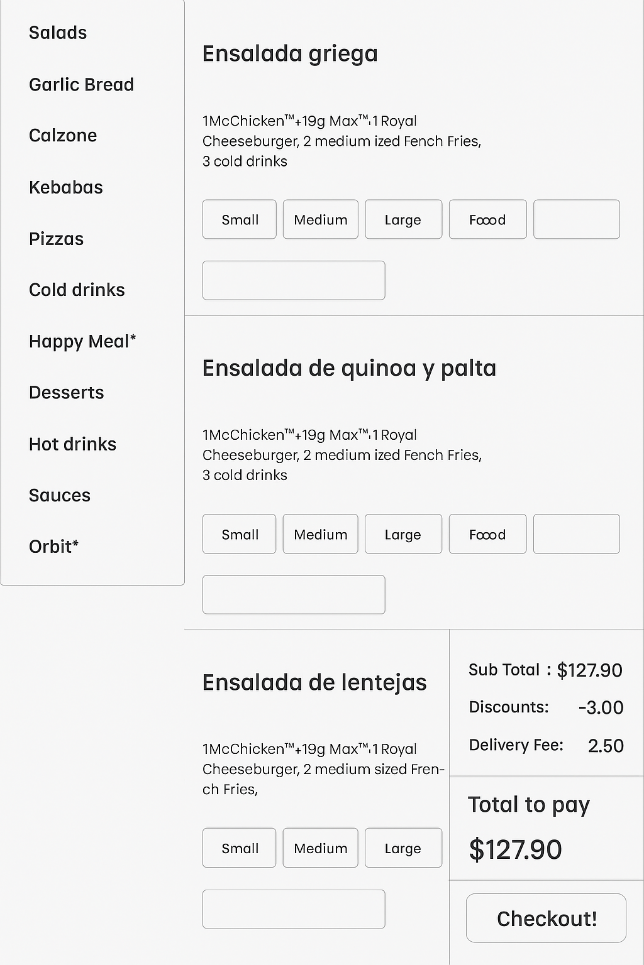
## UI / UX

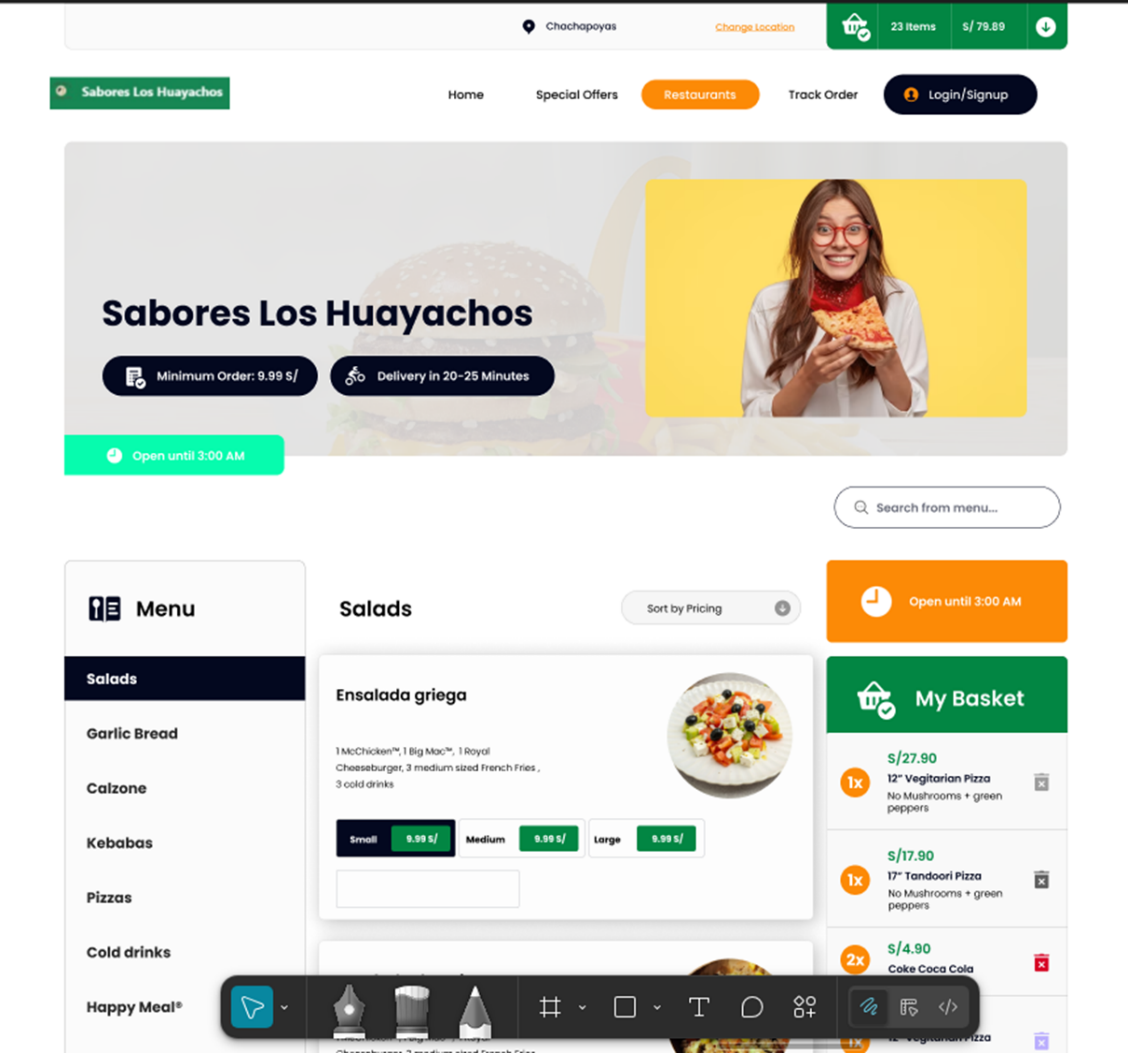
**Caso: Diseño UX/UI para aplicación web de pedidos saludables – Sabores Los Huayachos**

* Un inicio sencillo y atractivo: Los usuarios pueden visualizar promociones destacadas en la página principal y realizar pedidos de manera rápida con un solo clic.
* Un catálogo de productos organizado: La aplicación muestra los productos clasificados por categorías (ensaladas, bebidas, snacks, postres), con filtros para facilitar la búsqueda.
* Un carrito de compras intuitivo: Los clientes gestionan sus pedidos en tiempo real, pueden agregar, eliminar y confirmar productos fácilmente antes del pago.
* Un registro y login accesible: El sistema permite crear cuentas o iniciar sesión de forma rápida y segura, con opciones para recordar al usuario y recuperación de contraseña.
* Un seguimiento de pedido en tiempo real: Los clientes consultan el estado de sus pedidos (pendiente, en preparación, en camino, entregado) y pueden contactar al repartidor desde la misma interfaz.
* Un dashboard para el administrador: El administrador gestiona pedidos, asigna repartidores y visualiza métricas como ventas del día y productos más solicitados.

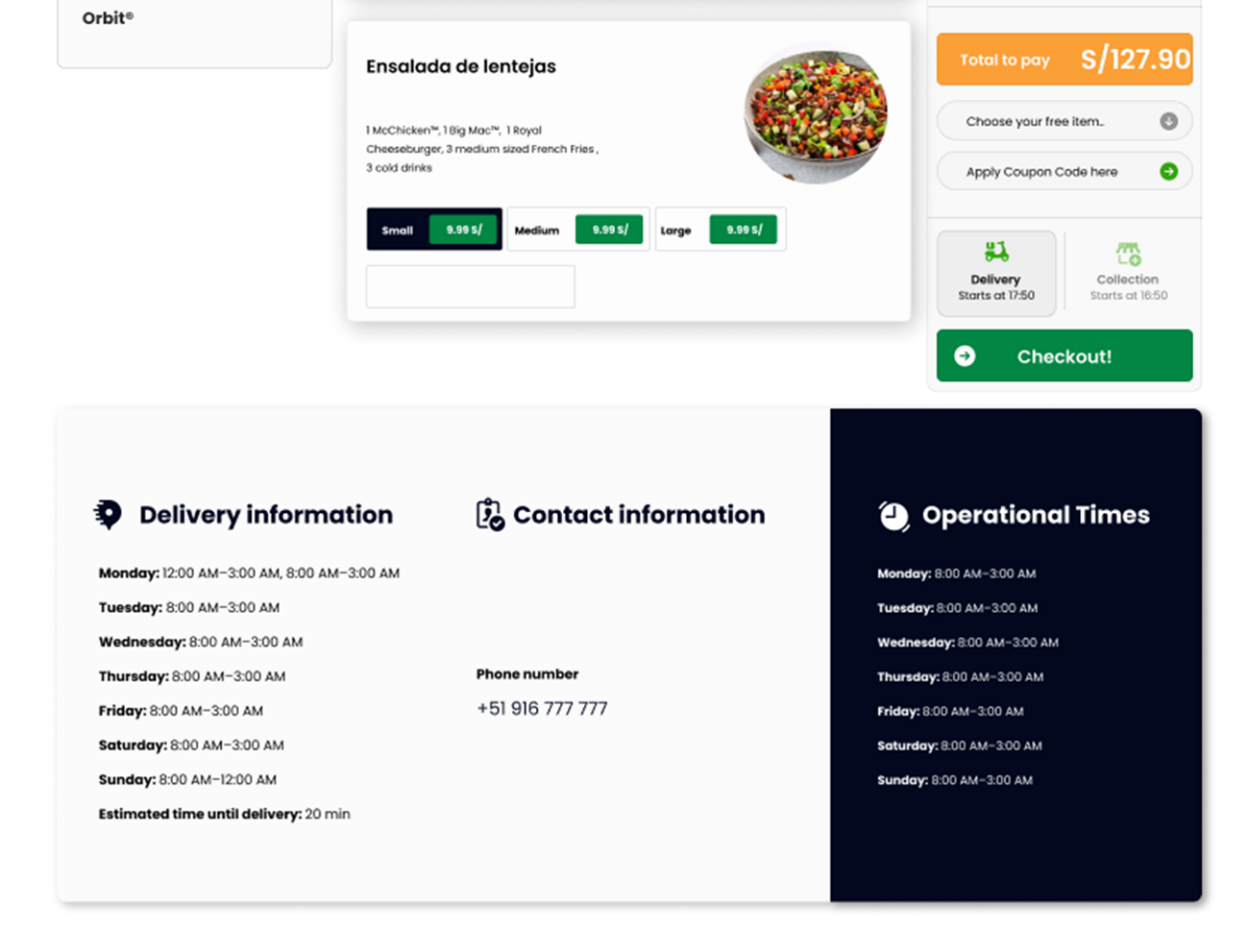
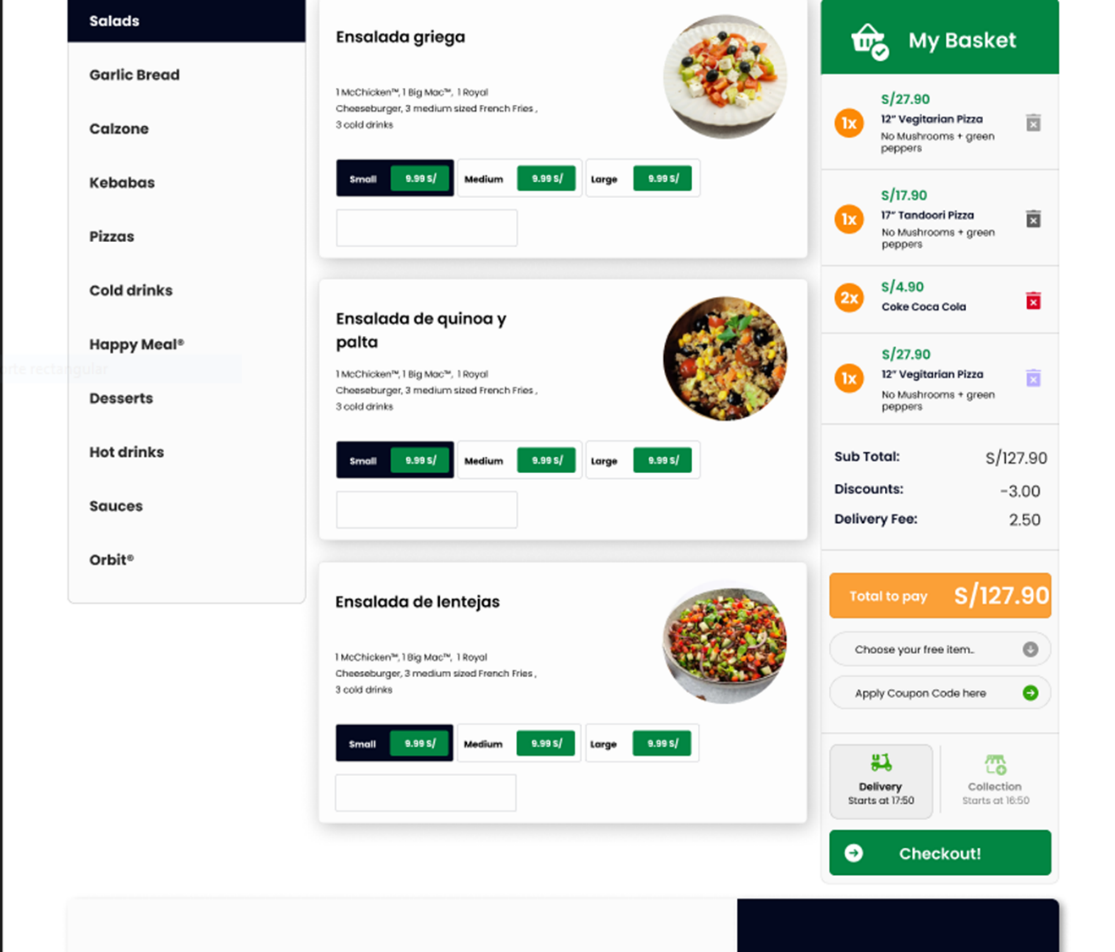
**Prototipos de bajo nivel y/o maquetado del producto a construir**





Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.



## Bibliografía

Bootstrap. (2023). Bootstrap 5 Documentation. Recuperado de [https://getbootstrap.com/docs/5.3/getting-started/introduction/](https://getbootstrap.com/docs/5.3/getting-started/introduction/?utm_source=chatgpt.com)

Java. (2024). The Java™ Tutorials. Oracle. Recuperado de <https://docs.oracle.com/javase/tutorial/>

Spring. (2024). Spring Boot Reference Documentation. Spring.io. Recuperado de <https://docs.spring.io/spring-boot/docs/current/reference/html/>

MySQL. (2024). MySQL 8.0 Reference Manual. Oracle. Recuperado de <https://dev.mysql.com/doc/refman/8.0/en/>

Mozilla Developer Network (MDN). (2024). JavaScript Guide. Recuperado de <https://developer.mozilla.org/es/docs/Web/JavaScript/Guide>

W3C. (2023). HTML Standard. World Wide Web Consortium. Recuperado de <https://html.spec.whatwg.org/>

Fernández, J. (2020). Desarrollo de aplicaciones web con PHP y MySQL: Desde cero hasta nivel profesional. Ediciones Marcombo.

Ullman, L. (2021). PHP and MySQL for Dynamic Web Sites: Visual QuickPro Guide (5th ed.). Peachpit Press.

Pressman, R. S., & Maxim, B. R. (2020). Ingeniería del software: Un enfoque práctico (8.ª ed.). McGraw-Hill.

Sommerville, I. (2016). Software Engineering (10th ed.). Pearson.