



Sistema ERP Completo - Documentación Actualizada



Actualización Reciente: Módulo de Productos



Nuevas Funcionalidades Implementadas



Módulo de Productos Avanzado con Múltiples Precios

El sistema ahora incluye un módulo completo de gestión de productos con soporte para **hasta 5 precios diferentes**, permitiendo mayor flexibilidad comercial.

Características del Producto:

Información Básica:

- Código único del producto
- Nombre y descripción detallada
- Código de barras
- Categoría y marca
- Modelo y presentación
- Contenido y peso
- Dimensiones, color, talla

Sistema de Múltiples Precios:

- **Precio 1:** Público general (obligatorio)
- **Precio 2:** Mayorista
- **Precio 3:** Distribuidor
- **Precio 4:** Especial/Corporativo
- **Precio 5:** Promocional/Liquidación
- Etiquetas personalizables para cada precio
- Cálculo automático de margen de ganancia

Control de Inventario:

- Stock actual, mínimo y máximo
- Unidades de medida configurables
- Alertas de stock bajo y sin stock
- Ubicación en almacén (pasillo, estante, nivel)
- Información del proveedor principal
- Tiempo de entrega

Características Especiales:

- Productos destacados y ofertas
- Control de vencimiento y lotes
- Productos controlados y con receta
- Galería de imágenes
- Estados activos/inactivos



Interfaz de Usuario del Módulo de Productos

Página Principal de Productos:

- Vista en tarjetas con información clave

- Búsqueda avanzada por múltiples campos
- Filtros por categoría, marca, estado de stock
- Indicadores visuales de estado (stock, destacado, oferta)
- Paginación inteligente
- Acciones rápidas (Ver, Editar, Eliminar)

Formulario de Producto:

Organizado en **5 pestañas** para mejor usabilidad:

1. **Básico:** Información general del producto
2. **Precios:** Configuración de los 5 precios y costos
3. **Inventario:** Control de stock y proveedores
4. **Ubicación:** Localización en almacén y lotes
5. **Avanzado:** Características especiales e imágenes

Vista Detallada del Producto:

- Información completa organizada por secciones
- Visualización de múltiples precios con colores distintivos
- Indicadores gráficos de stock con barras de progreso
- Estado visual del inventario
- Información completa de ubicación y vencimiento



APIs del Módulo de Productos

Endpoints Implementados:

1. **GET /api/productos**
 - Lista productos con filtros y paginación
 - Búsqueda por múltiples campos
 - Filtros por categoría, marca, estado
2. **POST /api/productos**
 - Creación de nuevos productos
 - Validación completa de datos
 - Verificación de códigos únicos
3. **GET /api/productos/[id]**
 - Obtener producto específico
 - Información completa con contadores
4. **PUT /api/productos/[id]**
 - Actualización de productos existentes
 - Validaciones de integridad de datos
5. **DELETE /api/productos/[id]**
 - Eliminación segura o desactivación
 - Protección de productos con transacciones
6. **GET /api/productos/categorias**
 - Lista de categorías disponibles
 - Para filtros y formularios

7. GET /api/productos/marcas

- Lista de marcas disponibles
- Para filtros y formularios

Actualización del Modelo de Base de Datos

Modelo Producto Expandido:

```

model Producto {
    id                String    @id @default(cuid())
    codigo            String    @unique
    nombre            String
    descripcion        String?
    categoria          String?
    marca              String?
    modelo             String?
    codigoBarras       String?

    // Información del producto
    presentacion       String?
    contenido           String?
    peso               Float?
    dimensiones         String?
    color              String?
    talla              String?

    // Múltiples precios (hasta 5 precios diferentes)
    precio1            Float    @default(0) // Precio público general
    precio2            Float    @default(0) // Precio mayorista
    precio3            Float    @default(0) // Precio distribuidor
    precio4            Float    @default(0) // Precio especial
    precio5            Float    @default(0) // Precio promocional

    // Etiquetas para los precios
    etiquetaPrecio1    String    @default("Público")
    etiquetaPrecio2    String    @default("Mayorista")
    etiquetaPrecio3    String    @default("Distribuidor")
    etiquetaPrecio4    String    @default("Especial")
    etiquetaPrecio5    String    @default("Promocional")

    // Costos y márgenes
    precioCompra       Float    @default(0)
    porcentajeGanancia Float    @default(0)

    // Inventario
    stock              Int       @default(0)
    stockMinimo        Int       @default(0)
    stockMaximo        Int       @default(1000)
    unidadMedida        String    @default("PZA")

    // Ubicación en almacén
    pasillo             String?
    estante             String?
    nivel              String?

    // Proveedores
    proveedorPrincipal String?
    tiempoEntrega       Int?     @default(0) // Días

    // Control de calidad
    fechaVencimiento    DateTime?
    lote                String?
    requiereReceta       Boolean @default(false)
    controlado           Boolean @default(false)

    // Multimedia
    imagen              String?
    imagenes            String[] // Array de URLs de imágenes

    // Control

```

```

isActive      Boolean @default(true)
destacado     Boolean @default(false)
oferta        Boolean @default(false)
createdAt     DateTime @default(now())
updatedAt     DateTime @updatedAt

// Relaciones
ventaItems    VentaItem[]
compraItems    CompraItem[]
movimientos    MovimientoInventario[]

@@map("productos")

```

Productos de Ejemplo Incluidos

El sistema incluye **7 productos de ejemplo** que demuestran la versatilidad del módulo:

1. **Laptop HP Pavilion 15** (Electrónicos)
 - 5 precios diferenciados
 - Información técnica completa
 - Producto destacado
2. **Samsung Galaxy A54** (Electrónicos)
 - Múltiples precios corporativos
 - Producto en oferta
 - Especificaciones técnicas
3. **Escritorio Ejecutivo Premium** (Muebles)
 - 3 precios comerciales
 - Información de peso y dimensiones
 - Tiempo de entrega extendido
4. **Café Premium Mexicano** (Alimentos)
 - 5 niveles de precio por volumen
 - Control de vencimiento
 - Información de lote
5. **Medicamento Paracetamol 500mg** (Farmacia)
 - Producto controlado
 - Precios institucionales
 - Control estricto de vencimiento
6. **Monitor LG 24" Full HD** (Electrónicos)
 - Precios especializados (gaming)
 - Especificaciones técnicas detalladas
7. **Aceite de Motor 20W-50** (Automotriz)
 - Precios por sector (talleres, flotillas)
 - Producto industrial

Sistema de Permisos Actualizado

Permisos para Módulo de Productos:

- **SUPERADMIN:** Acceso completo (crear, leer, actualizar, eliminar)
- **ADMIN:** Gestión completa excepto eliminación

- **ANALISTA:** Solo lectura
- **GESTOR:** Solo lectura
- **VENTAS:** Solo lectura
- **CLIENTE:** Solo lectura

Estructura de Permisos Actualizada:

```
productos: {
  read: boolean,
  create: boolean,
  update: boolean,
  delete: boolean
}
```

Arquitectura General del Sistema

Estructura del Proyecto

```
sistema_erp_completo/
├── app/
│   ├── app/
│   │   ├── productos/
│   │   │   ├── page.tsx
│   │   │   └── api/
│   │   │       ├── productos/
│   │   │       │   ├── route.ts
│   │   │       │   ├── [id]/route.ts
│   │   │       │   ├── categorias/route.ts
│   │   │       │   └── marcas/route.ts
│   │   │       ├── clientes/
│   │   │       ├── pagos/
│   │   │       ├── dashboard/
│   │   │       ├── clientes/
│   │   │       ├── dashboard/
│   │   │       ├── cobranza-movil/
│   │   │       ├── auth/
│   │   │       ├── components/
│   │   │       │   ├── productos/
│   │   │       │   │   ├── product-form.tsx
│   │   │       │   │   ├── product-details.tsx
│   │   │       │   │   └── product-filters.tsx
│   │   │       │   ├── navigation/
│   │   │       │   ├── ui/
│   │   │       │   └── dashboard/
│   │   │       ├── lib/
│   │   │       │   ├── auth.ts
│   │   │       │   ├── db.ts
│   │   │       │   ├── types.ts
│   │   │       │   ├── prisma/
│   │   │       │   │   ├── schema.prisma
│   │   │       │   │   └── scripts/
│   │   │       │   │       └── seed.ts
│   │   │       └── NUEVO: Módulo de Productos
│   │   │           ├── CRUD principal
│   │   │           ├── Operaciones por ID
│   │   │           ├── Lista de categorías
│   │   │           ├── Lista de marcas
│   │   │           ├── APIs de clientes
│   │   │           ├── APIs de pagos
│   │   │           ├── APIs del dashboard
│   │   │           ├── Gestión de clientes
│   │   │           ├── Dashboards
│   │   │           ├── Cobranza móvil
│   │   │           └── Autenticación
│   │   └── NUEVO: Componentes de Productos
│   │       ├── Formulario de productos
│   │       ├── Vista detallada
│   │       ├── Filtros avanzados
│   │       ├── Navegación
│   │       ├── Componentes UI base
│   │       └── Componentes dashboard
│   └── lib/
│       ├── auth.ts
│       ├── db.ts
│       ├── types.ts
│       ├── prisma/
│       │   ├── schema.prisma
│       │   └── scripts/
│       │       └── seed.ts
│       └── ACTUALIZADO: Tipos TypeScript
│           ├── ACTUALIZADO: Esquema de BD
│           └── ACTUALIZADO: Datos de prueba
```

Stack Tecnológico

Frontend:

- **Next.js 14** con App Router
- **React 18** con Server Components
- **TypeScript** para tipado estático
- **Tailwind CSS** para estilos
- **Radix UI** componentes accesibles
- **Framer Motion** para animaciones
- **React Hook Form** para formularios
- **Zod** para validaciones

Backend:

- **Next.js API Routes**
- **NextAuth.js** para autenticación
- **Prisma ORM** para base de datos
- **PostgreSQL** como base de datos
- **bcryptjs** para hashing de contraseñas

Funcionalidades PWA:

- **Service Worker** para cache offline
- **Web App Manifest** para instalación
- **Push Notifications** (configurado)
- **Background Sync** para cobranza móvil

Roles y Permisos del Sistema

Jerarquía de Roles:

1. **SUPERADMIN** - Control total del sistema
2. **ADMIN** - Gestión administrativa
3. **ANALISTA** - Análisis y reportes
4. **GESTOR** - Gestión de cobranza y clientes
5. **VENTAS** - Gestión de ventas y clientes
6. **CLIENTE** - Acceso limitado a su información

Matriz de Permisos:

Módulo	SUPER-ADMIN	ADMIN	ANALISTA	GESTOR	VENTAS	CLIENTE
Dashboard	✓	✓	✓	✓	✓	✗
Usuarios	CRUD	CRU	R	✗	✗	✗
Cientes	CRUD	CRU	R	CRU	CRU	✗
Productos	CRUD	CRU	R	R	R	R
Ventas	CRUD	CRU	R	R	CRU	R
Cobranza	CRUD	CRU	R	CRU	R	R
Almacén	CRUD	CRU	R	✗	R	✗
Reportes	CRU	CRU	CRU	R	R	✗
Configuración	CRU	R	R	✗	✗	✗

Leyenda: C=Create, R=Read, U=Update, D=Delete

**Modelo de Base de Datos Completo****Módulos Principales:****Autenticación y Usuarios**

- **User** - Usuarios del sistema con roles
- **Account** - Cuentas de OAuth
- **Session** - Sesiones de usuario
- **VerificationToken** - Tokens de verificación

**Gestión de Clientes**

- **Cliente** - Información completa de clientes
- **CreditoHistorial** - Historial crediticio

**Catálogo de Productos ✨ ACTUALIZADO**

- **Producto** - Productos con múltiples precios
- Soporte para hasta 5 precios diferenciados
- Control de inventario avanzado
- Información de ubicación en almacén

**Módulo de Ventas**

- **Venta** - Transacciones de venta
- **VentaItem** - Items de cada venta

Cobranza y Pagos

- Pago - Registro de pagos
- Sincronización offline/online
- Geolocalización de pagos

Inventario y Almacén

- Compra - Órdenes de compra
- CompraItem - Items de compra
- MovimientoInventario - Movimientos de stock
- Proveedor - Gestión de proveedores
- CuentaPorPagar - Cuentas por pagar

Configuración

- Configuracion - Configuración de marca blanca
- AuditLog - Auditoría del sistema

Funcionalidades del Sistema

Funcionalidades Implementadas

Autenticación y Seguridad

- Sistema de login con email/contraseña
- Roles de usuario con permisos granulares
- Sesiones seguras con NextAuth.js
- Middleware de autorización en todas las rutas

Gestión de Clientes

- CRUD completo de clientes
- Búsqueda avanzada e importación masiva
- Asignación de gestores y vendedores
- Vista detallada con historial
- Integración con WhatsApp y comunicación

Gestión de Productos ✨ **NUEVO**

- Múltiples precios por producto (hasta 5)
- Etiquetas personalizables para precios
- Control de inventario avanzado
- Ubicación en almacén
- Productos destacados y ofertas
- Control de vencimiento y lotes
- Búsqueda y filtros avanzados
- Vista detallada con visualización de precios

Cobranza Móvil (PWA)

- Búsqueda de clientes offline
- Registro de pagos con geolocalización
- Sincronización automática

- Impresión térmica Bluetooth
- Funcionalidad offline completa



Dashboard Analítico

- Métricas en tiempo real
- Estadísticas de ventas y cobranza
- Widgets específicos por rol
- Actividad reciente del sistema



Comunicación Integrada

- Envío de SMS masivos
- Integración con WhatsApp
- Mensajes automáticos de cobranza
- Notificaciones del sistema



Características Técnicas



APIs RESTful

- **23 endpoints** implementados
- Autenticación en todas las rutas
- Validación de permisos por rol
- Paginación y filtros avanzados
- Manejo de errores consistente



Progressive Web App (PWA)

- Instalable en dispositivos móviles
- Funcionalidad offline completa
- Service Worker para cache
- Manifest configurado
- Push notifications listas



Interfaz de Usuario

- Diseño responsivo para todos los dispositivos
- Componentes reutilizables con Radix UI
- Tema consistente con Tailwind CSS
- Animaciones suaves con Framer Motion
- Formularios reactivos con validación



Búsqueda y Filtros

- Búsqueda en tiempo real
 - Filtros combinables
 - Paginación eficiente
 - Ordenamiento personalizable
-

Configuración y Despliegue

Requisitos del Sistema

Desarrollo:

- Node.js 18 o superior
- Yarn como package manager
- PostgreSQL 14 o superior
- Git para control de versiones

Producción:

- Servidor con soporte Node.js
- Base de datos PostgreSQL
- SSL/HTTPS habilitado
- Dominio configurado

Instalación

```
# Clonar el repositorio
git clone [repository-url]
cd sistema_erp_completo/app

# Instalar dependencias
yarn install

# Configurar base de datos
npx prisma generate
npx prisma db push

# Ejecutar datos de ejemplo
npx tsx --require dotenv/config scripts/seed.ts

# Iniciar en desarrollo
yarn dev
```

Variables de Entorno

```
# Base de datos
DATABASE_URL="postgresql://..."

# Autenticación
NEXTAUTH_URL="http://localhost:3000"
NEXTAUTH_SECRET="your-secret-key"

# APIs externas (opcionales)
TWILIO_ACCOUNT_SID="..."
TWILIO_AUTH_TOKEN="..."
WHATSAPP_TOKEN="..."
```

Despliegue

```
# Compilar para producción
yarn build

# Iniciar servidor de producción
yarn start
```

Guía para Desarrolladores

Cómo Agregar Nuevos Módulos

1. Estructura de Archivos

```
app/
├── app/nuevo-modulo/
│   ├── page.tsx           # Página principal
│   └── api/nuevo-modulo/
│       ├── route.ts       # CRUD endpoints
│       └── [id]/route.ts   # Operaciones por ID
├── components/nuevo-modulo/
│   ├── form.tsx           # Formularios
│   ├── list.tsx           # Listados
│   └── details.tsx        # Vista detalle
```

2. Actualizar Permisos

```
// En lib/types.ts
export const RolePermissions = {
  SUPERADMIN: {
    // ... otros módulos
    nuevoModulo: { read: true, create: true, update: true, delete: true },
  },
  // ... otros roles
};
```

3. Agregar al Modelo de BD

```
// En prisma/schema.prisma
model NuevoModelo {
  id          String    @id @default(cuid())
  // ... campos del modelo
  createdAt   DateTime @default(now())
  updatedAt   DateTime @updatedAt





  @@map("nuevo_modulo")
}
```

4. Actualizar Navegación





```
// En components/navigation/sidebar.tsx
const navItems: NavItem[] = [
  // ... items existentes
  {
    title: 'Nuevo Módulo',
    href: '/nuevo-modulo',
    icon: IconComponent,
    permission: 'nuevoModulo',
  },
];
```

Mejores Prácticas





Seguridad:

-  Validar permisos en todas las APIs
-  Usar tipos TypeScript estrictos
-  Sanitizar inputs del usuario
-  Implementar rate limiting si es necesario

Performance:

-  Implementar paginación en listados
-  Usar React Query para cache
-  Optimizar consultas de base de datos
-  Implementar lazy loading

UX/UI:

-  Diseño responsivo obligatorio
-  Estados de carga y error
-  Validación de formularios en tiempo real
-  Feedback visual para acciones

Recursos y Enlaces

Documentación Técnica

- [Next.js Documentation](https://nextjs.org/docs) (https://nextjs.org/docs)
- [Prisma Documentation](https://prisma.io/docs) (https://prisma.io/docs)
- [NextAuth.js Documentation](https://next-auth.js.org) (https://next-auth.js.org)
- [Tailwind CSS Documentation](https://tailwindcss.com/docs) (https://tailwindcss.com/docs)

Herramientas de Desarrollo

- [Prisma Studio](https://prisma.io/studio) (https://prisma.io/studio) - Explorar base de datos
- [Next.js DevTools](https://nextjs.org/docs/advanced-features/debugging) (https://nextjs.org/docs/advanced-features/debugging)











Soporte

- **Usuarios de Prueba Disponibles:**
- `admin@sistema.com` / `123456` (SUPERADMIN)
- `johndoe.com` / `johndoe123` (SUPERADMIN)







- gestor1@sistema.com / password123 (GESTOR)
- vendedor1@sistema.com / password123 (VENTAS)

Registro de Cambios

v2.1.0 - Módulo de Productos ✨ ÚLTIMA ACTUALIZACIÓN





-  Implementado módulo completo de productos
-  Soporte para hasta 5 precios diferenciados
-  Control de inventario avanzado
-  Sistema de ubicación en almacén
-  Filtros y búsqueda avanzada
-  Vista detallada con visualización de precios
-  7 productos de ejemplo incluidos
-  APIs completas con validación de permisos
-  Actualización del sistema de permisos
-  Documentación completa actualizada

v2.0.0 - Sistema ERP Base





-  Módulo de gestión de clientes completo
-  Sistema de cobranza móvil (PWA)
-  Dashboard analítico
-  Sistema de autenticación con roles
-  Comunicación integrada (SMS/WhatsApp)
-  Base de datos PostgreSQL con Prisma

Próximas Funcionalidades

En Desarrollo:

-  Módulo de ventas completo
-  Sistema de reportes avanzados
-  Integración contable
-  Módulo de compras y proveedores

Planificadas:

-  App móvil nativa
-  Integración con facturación electrónica
-  Sistema de notificaciones push
-  Backup automático

Sistema ERP Completo - Gestión empresarial integral con tecnología moderna.

Última actualización: Septiembre 2025