

Documentación del Sistema ERP Comercial Integral

1. Información General del Proyecto

1.1 Descripción

Sistema ERP web multi-negocio que se adapta a diferentes verticales: comercio retail, talleres mecánicos, servicios profesionales, y más. Incluye gestión completa del ciclo comercial (compras y ventas), TPV offline, control de stock, proyectos y certificaciones.

1.2 Stack Tecnológico

- **Frontend:** Next.js 14+ (App Router)
 - **Framework UI:** React 18+
 - **Estilos:** Tailwind CSS + shadcn/ui
 - **Base de datos:** MongoDB
 - **ORM:** Mongoose
 - **Autenticación:** NextAuth.js
 - **Estado global:** Zustand
 - **BD Local TPV:** IndexedDB + Dexie.js
 - **Sincronización:** SWR + Service Workers
 - **Validación:** Zod
 - **Formularios:** React Hook Form
 - **PDF Generation:** React-PDF / jsPDF
 - **Comunicación en tiempo real:** Socket.io (opcional)
-

2. Arquitectura del Sistema

2.1 Estructura de Carpetas

/app

/(auth)

/login

/registro

/(dashboard)

/clientes

/proveedores

/productos

/variantes

/familias

/ventas

/presupuestos

/pedidos

/albaranes

/facturas

/compras

/presupuestos

/pedidos

/albaranes

/facturas

/tpv

/inventario

/movimientos

/almacenes

/transferencias

/taller (módulo específico)

/ordenes-trabajo

/vehiculos

/partes-trabajo

/proyectos

/tareas

/certificaciones

/seguimiento

/configuracion

/empresa

/series-documentos

/formas-pago

/tipos-negocio

/permisos

/api

/clientes

/proveedores

/productos

/ventas

/compras

/inventario

/taller
/proyectos
/sync (sincronización TPV)
/auth
/components
/ui
/layouts
/forms
/tables
/charts
/pdf-templates
/tpv
/lib
/db
 /mongodb
 /indexeddb
/sync
/utils
/validations
/pdf-generators
/hooks
/types
/models (schemas Mongoose)
/public
/workers (Service Workers)

2.2 Arquitectura Multinivel

- **Capa de Presentación:** Next.js + React
- **Capa de Negocio:** API Routes + Server Actions
- **Capa de Datos:** MongoDB (nube) + IndexedDB (local TPV)
- **Capa de Sincronización:** Background sync entre local y nube

3. Sistema Multi-Negocio

3.1 Tipos de Negocio Configurables

- **Retail/Comercio:** Tiendas, ecommerce
- **Taller Mecánico:** Reparación de vehículos
- **Servicios Profesionales:** Consultorías, agencias
- **Industria/Manufactura:** Producción y transformación
- **Hostelería:** Restaurantes, bares
- **Personalizado:** Configuración a medida

3.2 Configuración por Tipo de Negocio

Cada tipo de negocio activa/desactiva módulos:

```
javascript

{
  tipoNegocio: "taller_mecanico",
  modulosActivos: [
    "clientes",
    "vehiculos",
    "ordenes_trabajo",
    "partes_trabajo",
    "inventario",
    "compras",
    "ventas",
    "facturacion"
  ],
  camposPersonalizados: {
    cliente: ["vehiculos", "historial_reparaciones"],
    producto: ["aplicaciones_vehiculo", "referencias_fabricante"]
  }
}
```

4. Módulos Principales

4.1 Módulo de Autenticación y Usuarios

Funcionalidades:

- Sistema de roles jerárquico
- Permisos granulares por módulo
- Multi-empresa (SaaS)
- Autenticación 2FA (opcional)

Roles Predefinidos:

- Super Admin (multi-empresa)
 - Administrador (empresa)
 - Gerente
 - Vendedor
 - Técnico/Mecánico
 - Almacenero
 - Visualizador
-

4.2 Módulo de Clientes y Proveedores

4.2.1 Clientes

Campos Base:

- ID único
- Tipo (Particular, Empresa, Autónomo)
- Nombre/Razón Social
- NIF/CIF/DNI
- Email (múltiples)
- Teléfonos (múltiples)
- Direcciones (facturación, envío, múltiples)
- Persona de contacto
- Categoría (Minorista, Mayorista, VIP, etc.)
- Forma de pago preferida
- Plazo de pago
- Límite de crédito
- Descuento general (%)
- Observaciones
- Documentos adjuntos
- Estado (Activo/Inactivo/Bloqueado)

Campos Específicos Taller:

- Vehículos asociados
- Historial de reparaciones
- Preferencias de contacto
- Recordatorios de mantenimiento

Funcionalidades:

- CRUD completo
 - Importación/Exportación masiva
 - Fusión de clientes duplicados
 - Historial completo de interacciones
 - Portal del cliente (opcional)
-

4.3 Módulo de Productos y Variantes

4.3.1 Productos

Campos Base:

- ID/SKU único
- Referencia proveedor
- EAN/Código de barras
- Nombre
- Descripción corta
- Descripción larga
- Familia/Categoría (multinivel)
- Marca
- Unidad de medida
- IVA aplicable
- Tipo (Producto, Servicio, Combo)
- Estado (Activo, Descatalogado, Próximamente)

Gestión de Stock:

- Control de stock (Sí/No)
- Stock actual (por almacén)
- Stock mínimo
- Stock máximo
- Punto de pedido
- Stock reservado
- Stock disponible (calculado)
- Ubicación en almacén

Precios:

- Precio de compra
- Precio de venta (base)
- Tarifas personalizadas (por cliente/categoría)
- Precio con IVA
- Margen de beneficio (%)
- Historial de precios

Variantes:

- Sistema de atributos (Talla, Color, Material, etc.)
- Combinaciones automáticas
- SKU por variante
- Precio por variante
- Stock por variante
- Imágenes por variante

Ejemplo de Variantes:

javascript

```
{
  producto: "Camiseta Básica",
  atributos: [
    {
      nombre: "Talla",
      valores: ["XS", "S", "M", "L", "XL", "XXL"]
    },
    {
      nombre: "Color",
      valores: ["Blanco", "Negro", "Azul", "Rojo"],
      hexCodes: ["#FFFFFF", "#000000", "#0000FF", "#FF0000"]
    }
  ],
  variantes: [
    {
      sku: "CAM-BAS-XS-BLC",
      combinacion: { talla: "XS", color: "Blanco" },
      stock: 15,
      precioExtra: 0
    },
    // ... más variantes
  ]
}
```

Campos Específicos Taller:

- Tipo de pieza (Original, Alternativa, Recambio)
- Aplicaciones (marcas y modelos de vehículos)
- Referencias cruzadas
- Tiempo estimado de instalación
- Requiere mano de obra

Funcionalidades:

- Generador automático de variantes
- Importación masiva con variantes
- Gestión de familias/categorías ilimitadas
- Etiquetas y tags
- Imágenes múltiples
- Productos relacionados
- Kits y packs de productos

4.3.2 Familias y Categorías

- Estructura jerárquica ilimitada
 - Atributos personalizados por familia
 - Imágenes de categoría
 - Descuentos por familia
 - Comisiones por familia
-

4.4 Ciclo Comercial de VENTAS

El sistema sigue el flujo completo del ciclo de venta: **Presupuesto → Pedido → Albarán → Factura**

4.4.1 Presupuestos de Venta

Funcionalidades:

- Creación rápida desde plantillas
- Múltiples líneas de productos
- Variantes por línea
- Descuentos por línea y generales
- Validez del presupuesto
- Estados: Borrador, Enviado, Aceptado, Rechazado, Caducado
- Conversión a pedido con un clic
- Envío por email automático
- Versiones del presupuesto
- Margen de beneficio visible

Campos:

- Número de presupuesto (serie configurable)
- Fecha de emisión
- Fecha de validez
- Cliente
- Dirección de envío
- Condiciones de pago
- Forma de pago
- Líneas de detalle
- Subtotal
- Descuentos
- Base imponible
- IVA desglosado
- Total
- Observaciones
- Términos y condiciones

4.4.2 Pedidos de Venta**Funcionalidades:**

- Creación desde presupuesto o manual
- Estados: Pendiente, Confirmado, En preparación, Preparado, Enviado, Entregado, Cancelado
- Reserva automática de stock
- División en varios albaranes (entregas parciales)
- Seguimiento de estado
- Notificaciones al cliente

Campos adicionales:

- Fecha de entrega estimada
- Prioridad
- Estado de preparación
- Albaranes generados
- Pendiente de servir

4.4.3 Albaranes de Venta (Entregas)

Funcionalidades:

- Generación desde pedido
- Entregas parciales
- Control de cantidad servida vs pendiente
- Firma digital del receptor
- Geolocalización de entrega
- Generación de ruta de reparto
- Estados: Pendiente, En reparto, Entregado, Incidencia

Campos adicionales:

- Fecha y hora de entrega
- Transportista
- Número de bultos
- Peso total
- Receptor (nombre y firma)
- Observaciones de entrega

4.4.4 Facturas de Venta

Funcionalidades:

- Generación desde albaranes o pedidos
- Facturación masiva
- Múltiples series de facturación
- Rectificativas y abonos
- Vencimientos múltiples
- Estados: Emitida, Pagada parcial, Pagada, Vencida, Impagada
- Integración con contabilidad
- Envío automático a plataformas fiscales

Tipos de Factura:

- Factura normal
 - Factura simplificada
 - Factura rectificativa
 - Factura proforma
 - Abono/Nota de crédito
-

4.5 Ciclo Comercial de COMPRAS

Mismo flujo que ventas pero con proveedores: **Presupuesto → Pedido → Albarán → Factura**

4.5.1 Presupuestos de Compra

- Solicitud a múltiples proveedores
- Comparativa de presupuestos
- Aprobación de presupuestos
- Conversión a pedido

4.5.2 Pedidos de Compra

- Estados de seguimiento
- Fecha de recepción estimada
- Múltiples recepciones
- Alertas de retraso

4.5.3 Albaranes de Compra (Recepciones)

- Control de mercancía recibida
- Verificación de calidad
- Incidencias en recepción
- Entrada automática a inventario

4.5.4 Facturas de Compra

- Registro de facturas de proveedor
 - Control de vencimientos
 - Conciliación con albaranes
 - Estados de pago
-

4.6 TPV (Terminal Punto de Venta) con BD Local

4.6.1 Características Principales

Funcionamiento Offline:

- Base de datos local con IndexedDB
- Sincronización inteligente con servidor
- Cola de operaciones pendientes
- Resolución de conflictos

Datos Sincronizados Localmente:

- Productos completos con variantes
- Clientes frecuentes
- Familias y categorías
- Formas de pago
- Tarifas de precios
- Configuración del TPV
- Últimas 500 ventas

Funcionalidades:

- Búsqueda rápida de productos (código, nombre, EAN)
- Escaneo de código de barras
- Selección de variantes visual
- Calculadora de venta
- Múltiples formas de pago en una venta
- División de cuenta
- Apertura y cierre de caja
- Gestión de efectivo
- Devoluciones
- Tickets de espera
- Impresión de tickets
- Pantalla dual (cliente)

Interfaz TPV:

BÚSQUEDA	CARRITO	
PRODUCTOS		
	Producto 1	
[Grid Familias]	Producto 2	
[Grid Productos]		
	TOTAL: €XX.XX	
	[COBRAR]	

4.6.2 Sincronización

Estrategia:

- Sincronización automática cada X minutos (configurable)
- Sincronización manual on-demand
- Sincronización al abrir/cerrar TPV
- Sincronización de productos en background

Control de Conflictos:

- Timestamp de última modificación
- Resolución por prioridad (servidor gana)
- Log de conflictos para revisión manual

Arquitectura Sync:

```
javascript

// Service Worker para background sync
{
  productos: {
    lastSync: "2025-10-03T10:30:00Z",
    pending: [],
    changes: []
  },
  ventas: {
    pending: [/* ventas offline */],
    synced: [/* IDs sincronizados */]
  }
}
```

4.7 Módulo de Inventario

4.7.1 Almacenes

- Multi-almacén
- Ubicaciones dentro de almacén
- Zonas (picking, almacenaje, cuarentena)
- Responsables por almacén

4.7.2 Movimientos de Inventario

Tipos:

- Entrada por compra
- Salida por venta
- Traspaso entre almacenes
- Ajuste de inventario
- Producción/Transformación
- Merma
- Devolución cliente
- Devolución a proveedor

Control:

- Trazabilidad completa
- Lotes y números de serie
- Fechas de caducidad
- Auditoría de cambios

4.7.3 Inventario Físico

- Recuento de inventario
- Diferencias y ajustes
- Bloqueo de movimientos durante recuento
- Inventario rotativo
- Exportación para dispositivos móviles

4.7.4 Valoración de Inventario

- FIFO (First In, First Out)
- LIFO (Last In, First Out)
- Precio medio ponderado
- Precio última compra

4.8 Módulo Específico: Taller Mecánico

4.8.1 Vehículos

Campos:

- Matrícula
- Bastidor (VIN)
- Marca
- Modelo
- Año
- Color
- Kilometraje
- Combustible
- Cliente propietario
- Fecha próxima ITV
- Fecha próximo mantenimiento
- Historial completo
- Imágenes del vehículo
- Documentos (permisos, seguros)

4.8.2 Órdenes de Trabajo (OT)

Estructura:

- Número de OT
- Fecha de recepción
- Fecha de entrega estimada
- Fecha de entrega real
- Cliente
- Vehículo
- Kilometraje entrada
- Estado: Recibido, Diagnóstico, Presupuestado, Aprobado, En trabajo, Finalizado, Entregado
- Prioridad
- Técnico asignado

Diagnóstico:

- Descripción del problema
- Inspección visual
- Checklist configurable
- Fotos del vehículo
- Observaciones

Presupuesto Integrado:

- Líneas de mano de obra (horas \times tarifa)
- Líneas de repuestos
- Líneas de subcontrataciones
- Aprobación del cliente (firma digital)

Partes de Trabajo:

- División de la OT en tareas
- Asignación de técnicos
- Tiempo estimado vs real
- Control de tiempos
- Estado por tarea
- Comentarios del técnico

Finalización:

- Checklist de calidad
- Prueba de carretera
- Lavado del vehículo
- Generación de factura
- Entrega con firma

4.8.3 Seguimiento de Garantías

- Control de piezas en garantía
 - Fechas de vencimiento
 - Gestión de reclamaciones
-

4.9 Módulo de Proyectos

4.9.1 Gestión de Proyectos

Campos:

- Nombre del proyecto
- Cliente
- Descripción
- Fecha inicio
- Fecha fin estimada
- Estado: Propuesta, Aprobado, En curso, Pausado, Finalizado, Cancelado
- Responsable
- Equipo asignado
- Presupuesto
- Gastos reales
- Rentabilidad

Funcionalidades:

- Fases del proyecto
- Hitos (milestones)
- Entregables
- Gantt chart
- Control de costes
- Facturación por hitos

4.9.2 Tareas

- Asignación de tareas
- Dependencias entre tareas
- Tiempo estimado vs real
- Subtareas
- Comentarios
- Adjuntos
- Estados personalizables

4.9.3 Certificaciones

Para obras y proyectos con certificaciones:

- Certificaciones mensuales
 - Mediciones
 - Porcentaje de avance
 - Generación de facturas por certificación
 - Retenciones
 - Historial de certificaciones
-

4.10 Módulo de Reportes y Analytics

4.10.1 Dashboard Personalizable

- Widgets configurables
- KPIs en tiempo real
- Gráficos interactivos
- Filtros por fecha, almacén, vendedor, etc.

4.10.2 Reportes de Ventas

- Ventas por período
- Ventas por cliente
- Ventas por producto/familia
- Ventas por vendedor
- Análisis de márgenes
- Productos más vendidos
- Clientes más rentables
- Comparativas año anterior

4.10.3 Reportes de Compras

- Compras por período
- Compras por proveedor
- Análisis de costes
- Proveedores más utilizados

4.10.4 Reportes de Inventario

- Valoración de stock
- Rotación de productos
- Productos sin movimiento
- Stock mínimos alcanzados
- Necesidades de reposición
- Análisis ABC

4.10.5 Reportes Financieros

- Cuentas por cobrar (aging)
- Cuentas por pagar
- Flujo de caja proyectado
- Análisis de rentabilidad
- Balance de situación

4.10.6 Reportes Específicos Taller

- OTs por técnico
 - Tiempo medio de reparación
 - Vehículos atendidos
 - Servicios más solicitados
 - Análisis de tiempos
-

4.11 Módulo de Configuración

4.11.1 Datos de la Empresa

- Información fiscal
- Logos y marca
- Datos de contacto
- Configuración de impuestos
- Series de documentos

4.11.2 Series de Numeración

Configuración de series para cada documento:

- Prefijo (ejemplo: "FAC", "ALB", "PRES")
- Año en serie
- Contador actual
- Longitud de número
- Múltiples series por tipo

Ejemplo:

FAC-2025-00001
ALB-A-2025-00156
PRES-MADRID-2025-00042

4.11.3 Formas de Pago

- Efectivo
- Tarjeta
- Transferencia
- Bizum
- PayPal
- Financiación
- Pagaré
- Personalizado

Configuración por forma:

- Plazos de pago
- Vencimientos
- Comisiones
- Integración con TPV físico

4.11.4 Configuración por Tipo de Negocio

- Activar/desactivar módulos
- Campos personalizados
- Flujos de trabajo
- Plantillas de documentos
- Terminología personalizada

4.11.5 Usuarios y Permisos

Sistema de Permisos Granular:


```
javascript
```

```
{  
  modulo: "ventas",  
  permisos: {  
    presupuestos: {  
      ver: true,  
      crear: true,  
      editar: true,  
      eliminar: false,  
      aprobar: false  
    },  
    facturas: {  
      ver: true,  
      crear: false,  
      editar: false,  
      eliminar: false  
    }  
  }  
}
```

4.11.6 Plantillas de Documentos

- Editor visual de plantillas
- Variables dinámicas
- Múltiples plantillas por tipo
- HTML + CSS personalizable
- Vista previa en tiempo real

5. Modelos de Base de Datos (Mongoose Schemas)

5.1 Empresas (Multi-tenant)

javascript

```
{
  _id: ObjectId,
  nombre: String,
  nif: String,
  email: String,
  telefono: String,
  direccion: Object,
  logo: String,
  tipoNegocio: String,
  configuracion: Object,
  modulosActivos: [String],
  estado: String,
  fechaAlta: Date,
  plan: String
}
```

5.2 Usuarios

javascript

```
{
  _id: ObjectId,
  empresaId: ObjectId,
  email: String,
  password: String (hashed),
  nombre: String,
  apellidos: String,
  rol: String,
  permisos: Object,
  avatar: String,
  telefono: String,
  activo: Boolean,
  ultimoAcceso: Date,
  preferencias: Object,
  createdAt: Date,
  updatedAt: Date
}
```

5.3 Clientes

javascript

```
{
  _id: ObjectId,
  empresaId: ObjectId,
  tipo: String, // "particular" | "empresa" | "autonomo"
  codigo: String,
  nombre: String,
  nombreComercial: String,
  nif: String,
  emails: [String],
  telefonos: [String],
  direcciones: [{
    tipo: String, // "facturacion" | "envio"
    direccion: String,
    ciudad: String,
    provincia: String,
    codigoPostal: String,
    pais: String,
    predeterminada: Boolean
  }],
  personaContacto: String,
  categoria: String,
  formaPagoId: ObjectId,
  plazoPago: Number,
  limiteCredito: Number,
  descuentoGeneral: Number,
  tarifa: String,
  observaciones: String,
  documentosAdjuntos: [String],
  estado: String,
  // Campos específicos taller
  vehiculos: [ObjectId],
  // Campos personalizados
  camposPersonalizados: Object,
  createdAt: Date,
  updatedAt: Date
}
```

5.4 Proveedores

javascript

```
{
  _id: ObjectId,
  empresaId: ObjectId,
  codigo: String,
  nombre: String,
  nif: String,
  email: String,
  telefono: String,
  direccion: Object,
  personaContacto: String,
  formaPagoId: ObjectId,
  plazoPago: Number,
  descuentoGeneral: Number,
  observaciones: String,
  estado: String,
  createdAt: Date,
  updatedAt: Date
}
```

5.5 Familias

javascript

```
{
  _id: ObjectId,
  empresaId: ObjectId,
  codigo: String,
  nombre: String,
  descripcion: String,
  familiaPadreId: ObjectId,
  nivel: Number,
  imagen: String,
  atributosPersonalizados: [{
    nombre: String,
    tipo: String,
    obligatorio: Boolean,
    valores: [String]
  }],
  orden: Number,
  activo: Boolean,
  createdAt: Date,
  updatedAt: Date
}
```

5.6 Productos

javascript

```
{
  _id: ObjectId,
  empresaId: ObjectId,
  tipo: String, // "producto" | "servicio" | "combo"
  sku: String,
  referenciaProveedor: String,
  ean: String,
  nombre: String,
  descripcionCorta: String,
  descripcionLarga: String,
  familiaId: ObjectId,
  marca: String,
  unidadMedida: String,
  iva: Number,

  // Control de stock
  gestionStock: Boolean,
  stockMinimo: Number,
  stockMaximo: Number,
  puntoPedido: Number,

  // Precios
  precioCosto: Number,
  precioVenta: Number,
  margen: Number,
  tarifasPersonalizadas: [{
    nombre: String,
    precio: Number
  }],

  // Variantes
  tieneVariantes: Boolean,
  atributos: [{
    nombre: String,
    valores: [{
      valor: String,
      hexColor: String
    }]
  }],
  variantes: [{
    sku: String,
    combinacion: Object,
    precioExtra: Number,
    stock: Number,
    iva: Number,
    precioCosto: Number,
    precioVenta: Number,
    margen: Number,
    tarifasPersonalizadas: [{
      nombre: String,
      precio: Number
    }],
    atributos: [{
      nombre: String,
      valores: [{
        valor: String,
        hexColor: String
      }]
    }],
    tieneVariantes: Boolean
  }]
```

```

    imagenes: [String]
  }],

  // Stock por almacén
  stockPorAlmacen: [{
    almacenId: ObjectId,
    stock: Number,
    stockReservado: Number,
    ubicacion: String
  }],

  // Campos específicos taller
  tipoPieza: String,
  aplicacionesVehiculo: [{
    marca: String,
    modelos: [String]
  }],
  referenciasCruzadas: [String],
  tiempoInstalacion: Number,
  requiereManoObra: Boolean,

  // Multimedia
  imagenes: [String],
  documentos: [String],

  // Relacionados
  productosRelacionados: [ObjectId],
  kits: [{
    productoId: ObjectId,
    cantidad: Number
  }],

  // Meta
  etiquetas: [String],
  estado: String,
  camposPersonalizados: Object,
  createdAt: Date,
  updatedAt: Date
}

```

5.7 Almacenes

javascript

```
{
  _id: ObjectId,
  empresaId: ObjectId,
  codigo: String,
  nombre: String,
  direccion: Object,
  responsable: ObjectId,
  tipo: String, // "principal" | "secundario" | "transito"
  esTPV: Boolean,
  zonas: [{
    nombre: String,
    codigo: String
  }],
  activo: Boolean,
  createdAt: Date,
  updatedAt: Date
}
```

5.8 Presupuestos (Ventas/Compras)


```
{
  _id: ObjectId,
  empresaId: ObjectId,
  tipo: String, // "venta" | "compra"
  numero: String,
  serie: String,
  fecha: Date,
  fechaValidez: Date,

  // Partes involucradas
  clienteId: ObjectId, // o proveedorId
  clienteData: Object, // snapshot de datos
  direccionEnvio: Object,

  // Líneas
  lines: [{
    orden: Number,
    tipo: String, // "producto" | "servicio" | "texto"
    productoId: ObjectId,
    varianteId: String,
    descripcion: String,
    cantidad: Number,
    unidad: String,
    precioUnitario: Number,
    descuento: Number,
    iva: Number,
    subtotal: Number
  }],

  // Totales
  subtotal: Number,
  descuentoGeneral: Number,
  baseImponible: Number,
  totalIVA: Number,
  total: Number,

  // Desglose IVA
  desgloseIVA: [{
    porcentaje: Number,
    base: Number,
    cuota: Number
  }],

  // Condiciones
```

```
formaPagoId: ObjectId,  
plazoPago: Number,  
condicionesEntrega: String,  
observaciones: String,  
terminosCondiciones: String,  
  
// Estado  
estado: String, // "borrador" | "enviado" | "aceptado" | "rechazado" | "caducado"  
version: Number,  
  
// Conversión  
pedidoGenerado: ObjectId,  
  
// Auditoría  
creadoPor: ObjectId,  
modificadoPor: ObjectId,  
createdAt: Date,  
updatedAt: Date  
}
```

5.9 Pedidos (Ventas/Compras)

```
{
  _id: ObjectId,
  empresaId: ObjectId,
  tipo: String, // "venta" | "compra"
  numero: String,
  serie: String,
  fecha: Date,
  fechaEntregaEstimada: Date,
  fechaEntregaReal: Date,

  // Origen
  presupuestoId: ObjectId,

  // Partes involucradas
  clienteId: ObjectId, // o proveedorId
  clienteData: Object,
  direccionEnvio: Object,

  // Líneas con control de servicio
  lineas: [{
    orden: Number,
    productoId: ObjectId,
    varianteId: String,
    descripcion: String,
    cantidad: Number,
    cantidadServida: Number,
    cantidadPendiente: Number,
    unidad: String,
    precioUnitario: Number,
    descuento: Number,
    iva: Number,
    subtotal: Number,
    // Para compras: fecha recepción estimada por línea
    fechaRecepcionEstimada: Date
  }],

  // Totales
  subtotal: Number,
  descuentoGeneral: Number,
  baseImponible: Number,
  totalIVA: Number,
  total: Number,
  desgloseIVA: [{
    porcentaje: Number,
    base: Number,
    monto: Number,
    iva: Number
  }]
}
```

```

    base: Number,
    cuota: Number
  }},

  // Logística
  prioridad: String, // "baja" | "normal" | "alta" | "urgente"
  transportista: String,
  numeroSeguimiento: String,

  // Condiciones
  formaPagoId: ObjectId,
  plazoPago: Number,
  observaciones: String,
  observacionesInternas: String,

  // Estado y seguimiento
  estado: String, // "pendiente" | "confirmado" | "preparacion" | "preparado" | "enviado" | "entregado" | "cancelado"
  estadoPreparacion: Number, // porcentaje

  // Control de stock
  stockReservado: Boolean,

  // Relaciones
  albaranesGenerados: [ObjectId],
  facturasGeneradas: [ObjectId],

  // Auditoría
  creadoPor: ObjectId,
  modificadoPor: ObjectId,
  aprobadoPor: ObjectId,
  fechaAprobacion: Date,
  createdAt: Date,
  updatedAt: Date
}

```

5.10 Albaranes (Ventas/Compras)

javascript

```
{
  _id: ObjectId,
  empresaId: ObjectId,
  tipo: String, // "venta" | "compra"
  numero: String,
  serie: String,
  fecha: Date,
  fechaEntrega: Date,
  horaEntrega: String,

  // Origen
  pedidoId: ObjectId,

  // Partes involucradas
  clienteId: ObjectId, // o proveedorId
  clienteData: Object,
  direccionEnvio: Object,

  // Líneas entregadas
  lineas: [{
    orden: Number,
    lineaPedidoId: ObjectId,
    productoId: ObjectId,
    varianteId: String,
    descripcion: String,
    cantidadPedida: Number,
    cantidadEntregada: Number,
    unidad: String,
    lote: String,
    numeroSerie: String,
    fechaCaducidad: Date,
    almacenOrigen: ObjectId,
    ubicacion: String
  }],

  // Logística
  transportista: String,
  numeroSeguimiento: String,
  numeroBultos: Number,
  pesoTotal: Number,

  // Entrega
  receptorNombre: String,
  receptorDNI: String,
  fechaRecepcion: String, // De 6 a 14 caracteres
```

```
firmaReceptor: String, // Base64 imagen firma
geolocalizacion: {
  lat: Number,
  lng: Number
},
fotosEntrega: [String],
incidencias: String,

// Estado
estado: String, // "pendiente" | "reparto" | "entregado" | "incidencia"

// Facturación
facturaGenerada: ObjectId,
facturado: Boolean,

// Observaciones
observaciones: String,
observacionesInternas: String,

// Auditoría
creadoPor: ObjectId,
entregadoPor: ObjectId,
createdAt: Date,
updatedAt: Date
}
```

5.11 Facturas (Ventas/Compras)

javascript

```
{
  _id: ObjectId,
  empresaId: ObjectId,
  tipo: String, // "venta" | "compra"
  tipoFactura: String, // "normal" | "simplificada" | "rectificativa" | "proforma" | "abono"
  numero: String,
  serie: String,
  fecha: Date,
  fechaVencimiento: Date,

  // Origen
  pedidosId: [ObjectId],
  albaranesId: [ObjectId],

  // Si es rectificativa
  facturaRectificadaId: ObjectId,
  motivoRectificacion: String,

  // Partes involucradas
  clienteId: ObjectId, // o proveedorId
  clienteData: Object,
  direccionFacturacion: Object,

  // Líneas
  lineas: [{
    orden: Number,
    productoId: ObjectId,
    varianteId: String,
    descripcion: String,
    cantidad: Number,
    unidad: String,
    precioUnitario: Number,
    descuento: Number,
    iva: Number,
    recargoEquivalencia: Number,
    subtotal: Number,
    albaranReferencia: String
  }],

  // Totales
  subtotal: Number,
  descuentoGeneral: Number,
  baseImponible: Number,
  totalIVA: Number,
  totalBaseImponible: Number,
```

totalRecargoEquivalencia: Number,
retencionIRPF: Number,
total: Number,

// Desglose detallado

desgloseIVA: [{
 porcentaje: Number,
 base: Number,
 cuota: Number,
 recargoEquivalencia: Number
}],

// Pago

formaPagoId: ObjectId,
plazoPago: Number,
vencimientos: [{
 fecha: Date,
 importe: Number,
 pagado: Boolean,
 fechaPago: Date,
 metodoPago: String,
 referencia: String
}],

// Estado financiero

totalPagado: Number,
totalPendiente: Number,
estado: String, // "emitida" | "pagada_parcial" | "pagada" | "vencida" | "impagada"
diasVencimiento: Number,

// Contabilidad

asientoContable: String,
cuentaContable: String,

// Documentos

pdf: String,
xmlFacturacion: String,
enviada: Boolean,
fechaEnvio: Date,

// Observaciones

observaciones: String,
observacionesInternas: String,

// Auditoría

creadoPor: ObjectId,
aprobadoPor: ObjectId,


```
    createdAt: Date,  
    updatedAt: Date  
  }  
}
```

5.12 Movimientos de Inventario

javascript

```
{
  _id: ObjectId,
  empresaId: ObjectId,
  tipo: String, // "entrada" | "salida" | "traspaso" | "ajuste" | "produccion" | "merma" | "devolucion"
  fecha: Date,

  // Producto
  productoId: ObjectId,
  varianteId: String,
  descripcionProducto: String,

  // Almacenes
  almacenOrigenId: ObjectId,
  almacenDestinoId: ObjectId,
  ubicacionOrigen: String,
  ubicacionDestino: String,

  // Cantidad
  cantidad: Number,
  unidad: String,

  // Valoración
  precioUnitario: Number,
  valorTotal: Number,
  metodoValoracion: String, // "fifo" | "lifo" | "pmp"

  // Trazabilidad
  lote: String,
  numeroSerie: String,
  fechaCaducidad: Date,

  // Documento origen
  documentoTipo: String, // "venta" | "compra" | "traspaso" | "ajuste"
  documentoId: ObjectId,
  documentoNumero: String,

  // Información adicional
  motivo: String,
  observaciones: String,

  // Auditoría
  realizadoPor: ObjectId,
  aprobadoPor: ObjectId,
  createdAt: Date
}
```

}

5.13 Vehículos (Módulo Taller)

javascript

```
{
  _id: ObjectId,
  empresaId: ObjectId,
  clienteId: ObjectId,

  // Identificación
  matricula: String,
  bastidor: String,
  marca: String,
  modelo: String,
  version: String,
  año: Number,
  color: String,
  combustible: String, // "gasolina" | "diesel" | "hibrido" | "electrico" | "glp"
  transmision: String, // "manual" | "automatico"

  // Datos técnicos
  cilindrada: Number,
  potencia: Number,
  numeroMotor: String,
  kilometrajeActual: Number,

  // Documentación
  fechaMatriculacion: Date,
  fechaProximaITV: Date,
  fechaProximoMantenimiento: Date,
  numeroPoliza: String,
  companiaSeguro: String,
  fechaVencimientoSeguro: Date,

  // Multimedia
  imagenes: [String],
  documentos: [String],

  // Observaciones
  observaciones: String,
  accesorios: [String],

  // Historial
  ordenesTrabajoIds: [ObjectId],

  // Estado
  activo: Boolean,
  createdAt: Date,
  updatedAt: Date
```

updatedAt: Date

}

5.14 Órdenes de Trabajo (Módulo Taller)

javascript

```
{
  _id: ObjectId,
  empresaId: ObjectId,
  numero: String,
  serie: String,

  // Fechas
  fechaRecepcion: Date,
  fechaEntregaEstimada: Date,
  fechaEntregaReal: Date,
  horaRecepcion: String,

  // Cliente y vehículo
  clienteId: ObjectId,
  vehiculoId: ObjectId,
  vehiculoData: Object, // snapshot
  kilometrajeEntrada: Number,

  // Recepción
  nivelCombustible: String, // "vacio" | "1/4" | "1/2" | "3/4" | "lleno"
  estadoVehiculo: {
    carroceria: String,
    interior: String,
    neumaticos: String
  },
  objetosValor: [String],
  fotosRecepcion: [String],

  // Diagnóstico
  motivoVisita: String,
  descripcionProblema: String,
  diagnostico: String,
  checklistInspeccion: [{
    item: String,
    estado: String, // "ok" | "atencion" | "urgente"
    observaciones: String
  }],
  fotosDiagnostico: [String],

  // Presupuesto
  presupuestoAprobado: Boolean,
  firmaAprobacion: String,
  fechaAprobacion: Date,

  // Información adicional
```

// Líneas de trabajo

```
lineasManoObra: [{  
  descripcion: String,  
  tecnicoId: ObjectId,  
  horasEstimadas: Number,  
  horasReales: Number,  
  tarifaHora: Number,  
  subtotal: Number  
}],
```

```
lineasRepuestos: [{  
  productoId: ObjectId,  
  descripcion: String,  
  cantidad: Number,  
  precioUnitario: Number,  
  descuento: Number,  
  subtotal: Number,  
  aplicado: Boolean  
}],
```

```
lineasSubcontratacion: [{  
  descripcion: String,  
  proveedor: String,  
  importe: Number  
}],
```

// Totales

```
totalManoObra: Number,  
totalRepuestos: Number,  
totalSubcontratacion: Number,  
subtotal: Number,  
descuento: Number,  
baseImponible: Number,  
totalIVA: Number,  
total: Number,
```

// Partes de trabajo

```
partesTrabajo: [{  
  descripcion: String,  
  tecnicoId: ObjectId,  
  estado: String,  
  fechaInicio: Date,  
  fechaFin: Date,  
  tiempoEmpleado: Number,  
  observaciones: String,  
  fotos: [String]  
}],
```

// Estado

estado: String, // "recibido" | "diagnosis" | "presupuestado" | "aprobado" | "trabajo" | "finalizado" | "entregado" | "cancelado"
prioridad: String,

// Control de calidad

checklistCalidad: [{
 item: String,
 verificado: Boolean
}],
pruebaCarretera: Boolean,
lavado: Boolean,

// Entrega

entregadoA: String,
firmaEntrega: String,
fotosEntrega: [String],

// Facturación

facturaId: ObjectId,
facturado: Boolean,

// Seguimiento

notificacionesEnviadas: [{
 tipo: String,
 fecha: Date,
 metodo: String
}],

// Garantía

trabajosGarantia: Boolean,
referenciaGarantia: String,

// Observaciones

observaciones: String,
observacionesInternas: String,

// Auditoría

asesorId: ObjectId,
tecnicosAsignados: [ObjectId],
creadoPor: ObjectId,
modificadoPor: ObjectId,
createdAt: Date,
updatedAt: Date

}

5.15 Proyectos

javascript

```
{
  _id: ObjectId,
  empresaId: ObjectId,
  codigo: String,
  nombre: String,
  descripcion: String,

  // Cliente
  clienteId: ObjectId,
  personaContacto: String,

  // Fechas
  fechaInicio: Date,
  fechaFinEstimada: Date,
  fechaFinReal: Date,

  // Equipo
  responsableId: ObjectId,
  equipoIds: [ObjectId],

  // Fases del proyecto
  fases: [{
    nombre: String,
    descripcion: String,
    fechaInicio: Date,
    fechaFin: Date,
    estado: String,
    porcentajeCompletado: Number,
    entregables: [String]
  }],

  // Presupuesto y costes
  presupuestoTotal: Number,
  costosReales: Number,
  gastosAprobados: Number,
  rentabilidad: Number,
  margen: Number,

  // Facturación
  tipoFacturacion: String, // "hito" | "mensual" | "final" | "personalizado"
  facturado: Number,
  pendienteFacturar: Number,

  // Certificaciones (para obras)
```

```
certificaciones: [{  
  numero: Number,  
  fecha: Date,  
  periodo: String,  
  mediciones: Object,  
  porcentajeAvance: Number,  
  importeCertificado: Number,  
  retencion: Number,  
  facturaId: ObjectId,  
  aprobada: Boolean  
}],
```

```
// Control
```

```
porcentajeCompletado: Number,  
estado: String, // "propuesta" | "aprobado" | "curso" | "pausado" | "finalizado" | "cancelado"
```

```
// Hitos
```

```
hitos: [{  
  nombre: String,  
  descripcion: String,  
  fechaPrevista: Date,  
  fechaReal: Date,  
  completado: Boolean,  
  facturable: Boolean,  
  importe: Number  
}],
```

```
// Documentos
```

```
documentos: [{  
  nombre: String,  
  tipo: String,  
  url: String,  
  fechaSubida: Date  
}],
```

```
// Tareas relacionadas
```

```
tareasIds: [ObjectId],
```

```
// Observaciones
```

```
observaciones: String,  
riesgos: [String],
```

```
// Auditoría
```

```
creadoPor: ObjectId,  
modificadoPor: ObjectId,  
createdAt: Date,  
updatedAt: Date
```

}

5.16 Tareas

javascript

```
{
  _id: ObjectId,
  empresaId: ObjectId,
  proyectoId: ObjectId,
  titulo: String,
  descripcion: String,

  // Asignación
  asignadoA: ObjectId,
  creadoPor: ObjectId,

  // Fechas
  fechaInicio: Date,
  fechaVencimiento: Date,
  fechaCompletado: Date,

  // Tiempo
  tiempoEstimado: Number, // en horas
  tiempoReal: Number,

  // Prioridad y estado
  prioridad: String, // "baja" | "normal" | "alta" | "urgente"
  estado: String, // "pendiente" | "proceso" | "revision" | "completada" | "cancelada"

  // Dependencias
  tareasDependientes: [ObjectId],
  bloqueadaPor: [ObjectId],

  // Subtareas
  subtareas: [{
    titulo: String,
    completada: Boolean,
    asignadoA: ObjectId
  }],

  // Seguimiento
  progreso: Number, // 0-100

  // Comunicación
  comentarios: [{
    usuarioId: ObjectId,
    texto: String,
    fecha: Date
  }],
```

// Adjuntos

```
archivosAdjuntos: [{  
  nombre: String,  
  url: String,  
  tipo: String,  
  tamaño: Number,  
  fecha: Date  
}],
```

// Etiquetas

```
etiquetas: [String],
```

// Auditoría

```
createdAt: Date,  
updatedAt: Date  
}
```

5.17 Formas de Pago

javascript

```
{
  _id: ObjectId,
  empresaId: ObjectId,
  codigo: String,
  nombre: String,
  tipo: String, // "efectivo" | "tarjeta" | "transferencia" | "bizum" | "financiacion" | "pagare" | "personalizado"

  // Configuración
  plazoDias: Number,
  numeroPagos: Number,
  vencimientos: [{
    dias: Number,
    porcentaje: Number
  }],

  // Comisiones
  tieneComision: Boolean,
  comisionPorcentaje: Number,
  comisionFija: Number,

  // Integración TPV
  requiereTPVFisico: Boolean,
  configuracionTPV: Object,

  // Control
  activaParaVentas: Boolean,
  activaParaCompras: Boolean,
  predeterminada: Boolean,

  // Auditoría
  activo: Boolean,
  createdAt: Date,
  updatedAt: Date
}
```

5.18 Series de Numeración

javascript

```
{
  _id: ObjectId,
  empresaId: ObjectId,
  tipoDocumento: String, // "presupuesto_venta" | "pedido_venta" | "albaran_venta" | "factura_venta" | etc
  codigo: String,
  nombre: String,

  // Formato
  prefijo: String,
  incluirAño: Boolean,
  separador: String, // "-" | "/" | ""
  longitud: Number,
  contadorActual: Number,

  // Ejemplo: FAC-2025-00001
  formatoEjemplo: String,

  // Control
  predeterminada: Boolean,
  activa: Boolean,

  // Auditoría
  createdAt: Date,
  updatedAt: Date
}
```

5.19 Configuración de Empresa

javascript

```
{  
  _id: ObjectId,  
  empresaId: ObjectId,
```

// Datos fiscales

```
datosFiscales: {  
  razonSocial: String,  
  nombreComercial: String,  
  nif: String,  
  direccion: Object,  
  telefono: String,  
  email: String,  
  web: String,  
  registroMercantil: String  
},
```

// Branding

```
logos: {  
  principal: String,  
  secundario: String,  
  favicon: String  
},  
colorCorporativo: String,
```

// Impuestos

```
impuestos: {  
  ivaPorDefecto: Number,  
  recargoEquivalencia: Boolean,  
  retencionIRPF: Number  
},
```

// Configuración de negocio

```
tipoNegocio: String,  
modulosActivos: [String],
```

// Configuración de documentos

```
plantillasDocumentos: Object,  
terminosCondiciones: String,  
pieDocumentos: String,
```

// Notificaciones

```
notificaciones: {  
  emailStock: Boolean,  
  emailVencimientos: Boolean,
```

Client: Boolean


```
    smsClientes: Boolean
  },

  // Contabilidad
  planContable: String,
  ejercicioFiscal: {
    inicio: String, // "01-01"
    fin: String // "12-31"
  },

  // TPV
  configuracionTPV: {
    ticketPorDefecto: Boolean,
    impresora: String,
    mostrarPantallaDual: Boolean
  },

  // Integración
  apis: {
    contabilidad: Object,
    ecommerce: Object,
    email: Object,
    sms: Object
  },

  updatedAt: Date
}
```

5.20 Caja (TPV)

javascript

```
{
  _id: ObjectId,
  empresaId: ObjectId,
  almacenId: ObjectId,
  usuarioId: ObjectId,

  // Control de caja
  fechaApertura: Date,
  fechaCierre: Date,
  estado: String, // "abierta" | "cerrada"

  // Saldos
  saldoInicial: Number,
  efectivoInicial: Number,

  // Movimientos de efectivo
  movimientos: [{
    tipo: String, // "entrada" | "salida" | "venta"
    concepto: String,
    importe: Number,
    fecha: Date,
    ventaId: ObjectId
  }],

  // Ventas del turno
  ventasIds: [ObjectId],

  // Resumen por forma de pago
  resumen: [{
    formaPago: String,
    cantidad: Number,
    importe: Number
  }],

  // Cierre de caja
  efect
```