

Guía de Instalación - Sistema ERP Completo v4.0

Esta guía te ayudará a instalar y configurar el Sistema ERP Completo en tu entorno local o de producción.

Prerrequisitos

Software Requerido

- **Node.js** 18.0.0 o superior
- **npm** o **yarn** (recomendado yarn)
- **PostgreSQL** 14.0 o superior
- **Git** para clonar el repositorio

Herramientas Opcionales

- **Docker** y **Docker Compose** (para instalación containerizada)
- **VS Code** con extensiones de TypeScript y Prisma

Instalación Paso a Paso

Paso 1: Clonar el Repositorio

```
# Clonar el repositorio
git clone https://github.com/tu-usuario/sistema-erp-completo.git

# Navegar al directorio del proyecto
cd sistema-erp-completo/app
```

Paso 2: Instalar Dependencias

```
# Instalar dependencias con yarn (recomendado)
yarn install

# O con npm
npm install
```

Paso 3: Configurar Variables de Entorno

```
# Copiar el archivo de ejemplo
cp .env.example .env

# Editar el archivo .env con tus configuraciones
nano .env
```

Configuraciones mínimas requeridas:

```
# Base de datos
DATABASE_URL="postgresql://usuario:password@localhost:5432/erp_database"

# Autenticación
NEXTAUTH_URL="http://localhost:3000"
NEXTAUTH_SECRET="tu_secret_super_seguro"

# Configuración de empresa
COMPANY_NAME="Tu Empresa S.A. de C.V."
COMPANY_RFC="XAXX010101000"
```

Paso 4: Configurar Base de Datos PostgreSQL

Opción A: Instalación Local

1. Instalar PostgreSQL

```
# Ubuntu/Debian
sudo apt update
sudo apt install postgresql postgresql-contrib

# macOS con Homebrew
brew install postgresql
```

1. Crear base de datos

```
# Conectar a PostgreSQL
sudo -u postgres psql

# Crear base de datos y usuario
CREATE DATABASE erp_database;
CREATE USER erp_user WITH ENCRYPTED PASSWORD 'tu_password';
GRANT ALL PRIVILEGES ON DATABASE erp_database TO erp_user;
\q
```

Opción B: Docker (Recomendado para desarrollo)

```
# Ejecutar PostgreSQL en Docker
docker run --name postgres-erp \
  -e POSTGRES_DB=erp_database \
  -e POSTGRES_USER=erp_user \
  -e POSTGRES_PASSWORD=tu_password \
  -p 5432:5432 \
  -d postgres:14
```

Paso 5: Configurar Prisma y Base de Datos

```
# Generar el cliente de Prisma
yarn prisma generate

# Aplicar el esquema a la base de datos
yarn prisma db push

# Opcional: Sembrar datos de prueba
yarn prisma db seed
```

Paso 6: Ejecutar en Desarrollo

```
# Iniciar el servidor de desarrollo
yarn dev

# La aplicación estará disponible en:
# http://localhost:3000
```

Instalación con Docker

Docker Compose (Más fácil)

1. Crear archivo docker-compose.yml

```
version: '3.8'
services:
  postgres:
    image: postgres:14
    environment:
      POSTGRES_DB: erp_database
      POSTGRES_USER: erp_user
      POSTGRES_PASSWORD: erp_password
    ports:
      - "5432:5432"
    volumes:
      - postgres_data:/var/lib/postgresql/data

  app:
    build: .
    ports:
      - "3000:3000"
    environment:
      DATABASE_URL: "postgresql://erp_user:erp_password@postgres:5432/erp_database"
      NEXTAUTH_URL: "http://localhost:3000"
      NEXTAUTH_SECRET: "tu_secret_aqui"
    depends_on:
      - postgres

volumes:
  postgres_data:
```

1. Ejecutar con Docker Compose

```
# Construir e iniciar los servicios
docker-compose up --build

# En modo detached
docker-compose up -d --build
```

Configuración de Autenticación

Generar NEXTAUTH_SECRET

```
# Generar un secret seguro
openssl rand -base64 32

# O usar el generador online:
# https://generate-secret.vercel.app/32
```

Configurar Proveedores de Auth (Opcional)

```
# Google OAuth
GOOGLE_CLIENT_ID="tu_google_client_id"
GOOGLE_CLIENT_SECRET="tu_google_client_secret"

# GitHub OAuth
GITHUB_ID="tu_github_client_id"
GITHUB_SECRET="tu_github_client_secret"
```

Configurar Integración de Pagos (Opcional)

OpenPay (Recomendado para México)

1. **Crear cuenta en OpenPay**
 - Registro: <https://sandbox.openpay.mx>
 - Obtener credenciales de prueba
2. **Configurar variables de entorno**

```
OPENPAY_MERCHANT_ID="tu_merchant_id"
OPENPAY_PUBLIC_KEY="tu_public_key"
OPENPAY_PRIVATE_KEY="tu_private_key"
OPENPAY_ENVIRONMENT="sandbox"
```

Configurar PWA (Opcional)

El sistema incluye funcionalidad PWA lista para usar:

1. **Generar iconos de PWA**
 - Colocar iconos en `/public/icons/`
 - Formatos: 192x192 y 512x512 PNG
2. **Personalizar manifest.json**

```
{
  "name": "Tu Sistema ERP",
  "short_name": "ERP",
  "description": "Sistema ERP Completo para tu empresa",
  "start_url": "/",
  "display": "standalone"
}
```

Optimizaciones de Producción

Paso 1: Variables de Entorno de Producción

```
NODE_ENV="production"
DATABASE_URL="postgresql://user:pass@prod-server:5432/erp_prod"
NEXTAUTH_URL="https://tudominio.com"
NEXTAUTH_SECRET="secret_super_seguro_para_produccion"
```

Paso 2: Build de Producción

```
# Crear build optimizado
yarn build

# Iniciar en modo producción
yarn start
```

Paso 3: Configurar Servidor Web

Nginx (Recomendado)

```
server {
    listen 80;
    server_name tudominio.com;

    location / {
        proxy_pass http://localhost:3000;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
    }
}
```



Solución de Problemas Comunes

Error de Conexión a Base de Datos

```
# Verificar conexión
yarn prisma db push --preview-feature

# Reset completo de BD (;CUIDADO EN PRODUCCIÓN!)
yarn prisma db reset
```

Error de Dependencias

```
# Limpiar cache e instalar
rm -rf node_modules yarn.lock
yarn install
```

Error de Build

```
# Verificar TypeScript
yarn tsc --noEmit

# Limpiar cache de Next.js
rm -rf .next
yarn build
```

Error de Prisma

```
# Regenerar cliente
yarn prisma generate

# Verificar esquema
yarn prisma validate
```



Verificar Instalación

Checklist Post-Instalación

- [] ☒ La aplicación inicia sin errores (`yarn dev`)
- [] ☒ La base de datos se conecta correctamente
- [] ☒ La página de login es accesible
- [] ☒ Se pueden crear usuarios
- [] ☒ El dashboard principal carga
- [] ☒ Los módulos principales funcionan
- [] ☒ Los reportes generan datos

URLs de Verificación

```
# Página principal
http://localhost:3000

# API de salud
http://localhost:3000/api/health

# Página de login
http://localhost:3000/auth/signin

# Dashboard
http://localhost:3000/dashboard
```



Próximos Pasos

Después de una instalación exitosa:

1. Configurar tu empresa

- Ir a `/configuracion`
- Configurar datos de la empresa
- Personalizar colores y logo

2. Crear usuarios

- Configurar roles y permisos
- Invitar usuarios del equipo

3. Importar datos

- Catálogo de productos
- Base de clientes
- Proveedores

4. Configurar integraciones

- Pasarela de pagos
- Sistema de SMS/WhatsApp
- Facturación electrónica

Obtener Ayuda

- **Documentación:** Revisar archivos en `/docs`
 - **Issues:** Crear issue en el repositorio de GitHub
 - **Estado del proyecto:** Ver `PROYECTO_STATUS.md`
 - **Changelog:** Revisar `CHANGELOG_v4.md`
-

¡Felicidades! Tu Sistema ERP está listo para usar 🎉

Desarrollado con ❤️ usando DeepAgent de Abacus.AI 🚀