





Manual de Instalación y Ejecución - Simulación de Tienda

1 Requisitos Previos

Antes de ejecutar el proyecto, asegúrate de tener instalados los siguientes programas en tu sistema:

Software Necesario

- **Node.js** (versión 16 o superior)  [Descargar aquí](#)
- **MongoDB**  [Descargar aquí](#)
- **SQL Server**  [Descargar aquí](#)
- **Git**  [Descargar aquí](#)

Configuración de Base de Datos

1. **SQL Server:** Asegúrate de que **SQL Server** **está en ejecución** y puedes conectarte con Sequelize.
2. **MongoDB:** Ejecuta MongoDB localmente (mongod) o con un servicio en la nube como **MongoDB Atlas**.

2 Clonación del Proyecto

Para obtener el código fuente, clona el repositorio desde GitHub:

```
git clone https://github.com/martinheler/Tienda.git
```

```
cd Tienda
```

3 Configuración del Proyecto

Instalar dependencias

Ejecuta el siguiente comando dentro de la carpeta del proyecto:

```
npm install
```

Esto instalará todas las dependencias necesarias.

Configurar Variables de Entorno (.env)

Crea un archivo **.env** en la raíz del proyecto y agrega la configuración:

```
PORT=3000
```

```
SQL_SERVER_HOST=localhost
```

SQL_SERVER_USER=sa

SQL_SERVER_PASSWORD=tu_contraseña

SQL_SERVER_DATABASE=tienda

MONGODB_URL=mongodb://localhost:27017/tienda_logs

JWT_SECRET=supersecretkey

⚠ IMPORTANTE:

Si usas MongoDB Atlas, cambia MONGODB_URL con tu **cadena de conexión de Atlas**.

◆ 4 Crear Base de Datos

🚀 Ejecutar migraciones de SQL Server

Ejecuta el siguiente comando para crear las tablas en SQL Server:

```
npx sequelize-cli db:migrate
```

Si necesitas reiniciar las tablas:

```
npx sequelize-cli db:migrate:undo:all
```

```
npx sequelize-cli db:migrate
```

◆ 5 Ejecutar el Proyecto

Ejecuta el siguiente comando para iniciar la API:

```
npm run dev
```

Si deseas compilar y ejecutar en producción:

```
npm run build
```

```
npm start
```

El servidor se ejecutará en **http://localhost:3000** 🚀

◆ 6 Endpoints Disponibles

🚀 CRUD de Productos

Método	Ruta	Descripción
GET	/products	Obtener todos los productos
POST	/products	Crear un producto
PUT	/products/:id	Actualizar un producto
DELETE	/products/:id	Eliminar un producto

Registro y Autenticación de Usuarios

Método	Ruta	Descripción
POST	/users/register	Registrar un usuario
POST	/users/login	Iniciar sesión

Creación y Consulta de Pedidos

Método	Ruta	Descripción
POST	/orders	Crear un pedido
GET	/orders/:userId	Obtener pedidos de un usuario

Reportes en Vivo

Método	Ruta	Descripción
GET	/reports/daily-sales	Total de ventas del día
GET	/reports/best-selling-product	Producto más vendido
GET	/reports/top-users	Usuarios con más compras

 Los reportes se actualizan en tiempo real con socket.io.

Pruebas Automáticas

Para ejecutar las pruebas, usa:

```
npm test
```

