

**AutoMax**

Documentación Oficial del Proyecto Fullstack

Autor: Roberto Anania

Fecha: 17-12-2025

Institución: DUOC UC

# **Índice**

Clic derecho sobre el índice → Actualizar campo.

Tabla de contenido (actualizar en Word)

# **1. Introducción**

El presente documento describe formalmente el proyecto AutoMax, una solución fullstack orientada a la gestión integral de talleres automotrices.

# **2. Alcance y objetivos**

• Centralizar la administración de turnos, vehículos, repuestos y servicios.

• Proveer un portal por roles: Administrador y Cliente.

• Exponer una API REST segura mediante JWT para consumo del frontend.

# **3. Arquitectura**

## **3.1 Vista general**

El sistema se compone de un frontend web (React) y un backend (Spring Boot) que expone una API REST. La persistencia se realiza en MongoDB Atlas.

## **3.2 Tecnologías**

• Frontend: React, React Router, React Bootstrap, Axios.

• Backend: Spring Boot, Spring Security, JWT, Spring Data MongoDB.

• Base de datos: MongoDB Atlas.

# **4. ERS - Especificación de Requisitos**

## **4.1 Roles**

• ADMIN: acceso completo a gestión operativa y administración del sistema.

• CLIENTE: acceso a funcionalidades orientadas a sus vehículos y turnos.

## **4.2 Requisitos funcionales**

• Registro e inicio de sesión de usuarios.

• Autorización basada en roles y token JWT.

• Gestión de agenda/turnos.

• Gestión de vehículos e historial.

• Gestión de servicios y repuestos.

• Gestión de órdenes y mensajería interna.

• Gestión de usuarios (solo ADMIN).

## **4.3 Requisitos no funcionales**

• Seguridad: JWT, control de acceso por roles y CORS restringido.

• Mantenibilidad: separación por capas (controller/service/repository).

• Usabilidad: interfaz consistente y navegación simple.

• Disponibilidad: scripts de inicio/detención y mitigación de problemas de DNS en redes móviles.

# **5. Manual de usuario**

## **5.1 Acceso**

• Frontend: http://localhost:3000

• Backend: http://localhost:8080

• API Docs: http://localhost:8080/api-docs

## **5.2 Credenciales de prueba**

• Administrador: admin@automax.cl / admin123

• Cliente: cliente@test.cl / cliente123

# **6. Testing y cobertura**

La cobertura se valida mediante checklist de pruebas funcionales sobre flujos críticos del sistema (en contexto académico).

## **6.1 Checklist**

• Login ADMIN/CLIENTE exitoso y persistencia de sesión (token).

• Validación de permisos por rol.

• CRUD de agenda/turnos y vehículos.

• CRUD de servicios y repuestos.

• Gestión de usuarios (ADMIN): crear/editar/cambiar contraseña/eliminar.

# **7. Documentación de APIs**

• Base URL: http://localhost:8080/api

• Autenticación: Authorization: Bearer <token>

## **7.1 Endpoints**

### **Agenda**

• GET /api/agenda

• POST /api/agenda

• PUT /api/agenda/{id}

• DELETE /api/agenda/{id}

### **Auth**

• POST /api/auth/register

• POST /api/auth/login

### **Mensaje**

• GET /api/mensajes

• POST /api/mensajes

• PUT /api/mensajes/{id}/destacado

• DELETE /api/mensajes/{id}

### **Orden**

• GET /api/ordenes

• GET /api/ordenes/{id}

• POST /api/ordenes

• DELETE /api/ordenes/{id}

### **Repuesto**

• GET /api/repuestos

• POST /api/repuestos

• PUT /api/repuestos/{id}

• DELETE /api/repuestos/{id}

### **Servicio**

• GET /api/servicios

• GET /api/servicios/{id}

• POST /api/servicios

• PUT /api/servicios/{id}

• DELETE /api/servicios/{id}

### **Usuario**

• GET /api/usuarios

• GET /api/usuarios/{id}

• POST /api/usuarios

• PUT /api/usuarios/{id}

• PUT /api/usuarios/{id}/password

• DELETE /api/usuarios/{id}

### **Vehiculo**

• GET /api/vehiculos

• POST /api/vehiculos

• POST /api/vehiculos/{id}/historial

• PUT /api/vehiculos/{id}

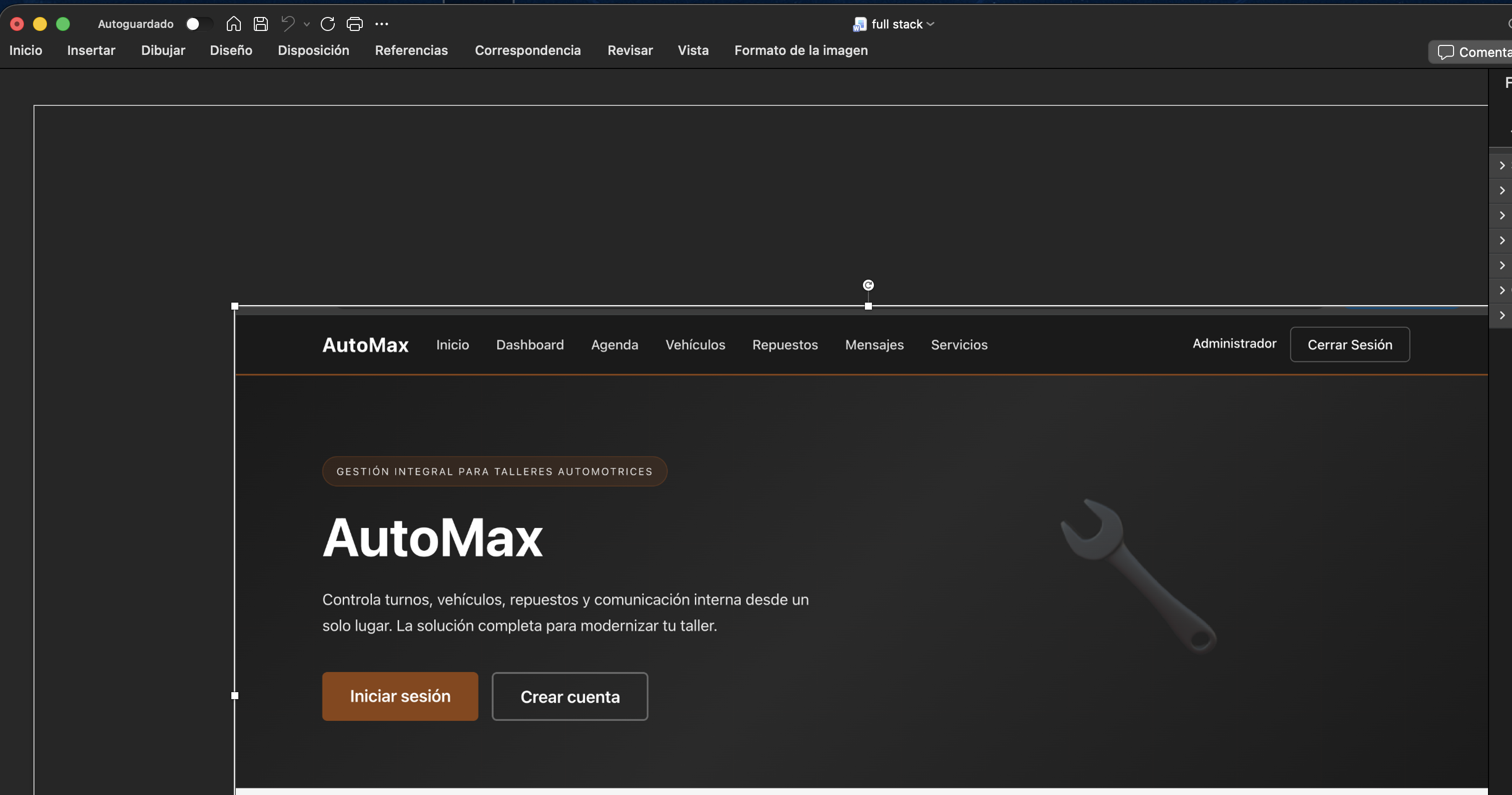
• DELETE /api/vehiculos/{id}

# **8. API e integración**

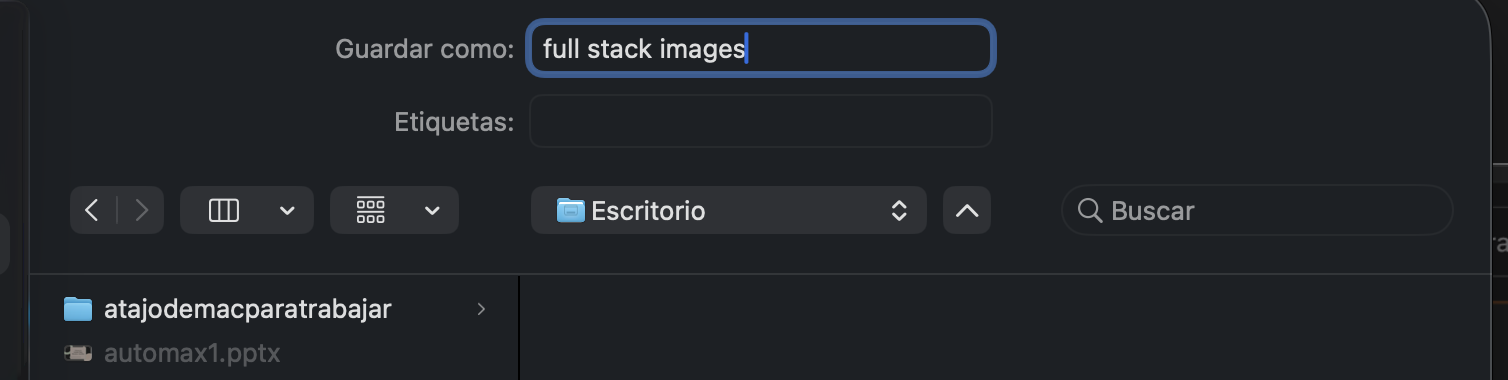
El frontend consume la API mediante Axios, adjuntando el token JWT en cada solicitud. Ante error 401, la sesión se invalida y se redirige al inicio de sesión.

# **9. Anexos - Capturas**

**Figura 1. Evidencia visual de funcionamiento**



**Figura 2. Evidencia visual de funcionamiento**



**Figura 3. Evidencia visual de funcionamiento**

