

Plataforma Multicanal de Atención al Cliente

Prueba Técnica - Next.js + WhatsApp Business API

Versión 1.0

<https://tusitioejemplo.com>

Índice

1. Objetivo General
2. Requisitos Técnicos
3. Funcionalidades Principales
4. Bonus (Opcionales)
5. Entregables
6. Tiempo Estimado

Prueba Técnica - Plataforma Multicanal de Atención al Cliente

1. Objetivo General

Desarrollar una aplicación SaaS multicitiente tipo "Helpdesk", donde empresas puedan crear una cuenta, conectar su número de WhatsApp Business y gestionar conversaciones con clientes desde un panel web.

La plataforma debe construirse con Next.js (App Router) e incluir autenticación, rutas dinámicas por tenant, integración con la API de WhatsApp Business, base de datos relacional y una interfaz moderna y responsive.

2. Requisitos Técnicos

- Multitenencia por empresa (subdominios o rutas tipo /empresa).
- Autenticación con next-auth (JWT y/o OAuth).
- Rutas protegidas con middleware de Next.js.
- Combinación de SSR y SSG.
- API Routes y Server Actions para integración backend.
- Base de datos relacional con Prisma o Drizzle.
- UI con TailwindCSS.
- Webhooks y conexión a la Cloud API de Meta.
- Uso de revalidación, SWR o React Query para datos dinámicos.

3. Funcionalidades Principales

- Registro/Login de empresas.
- Dashboard con conversaciones activas.
- Ver y responder mensajes en tiempo real (integración con WhatsApp).
- Webhook para recibir mensajes entrantes.
- Asignar etiquetas a conversaciones.
- Gestión de plantillas de respuesta (Templates API).
- Vista del historial del cliente.

4. Bonus (Opcionales)

Prueba Técnica - Plataforma Multicanal de Atención al Cliente

- Soporte multiagente por empresa.
- Panel de métricas: tiempo de respuesta, satisfacción.
- Modo oscuro.
- Funcionalidad demo.
- Internacionalización con i18n.
- Roles por usuario (admin, agente, solo lectura).

5. Entregables

- Repositorio público en GitHub.
- README con instrucciones de instalación y ejecución.
- Documento técnico explicando decisiones de arquitectura y herramientas utilizadas.

6. Tiempo Estimado

La prueba puede completarse en un plazo estimado de 20 a 25 horas, distribuidas en varios días.

Se evaluará calidad del código, claridad en la arquitectura y buena experiencia de usuario.