

Análisis de Implementación: Chat Inteligente para Guías de Huéspedes

1. ARQUITECTURA DE LA SOLUCIÓN

Stack Tecnológico Propuesto

Frontend & Backend:

- Next.js 14+ (App Router) - ya existente
- React Server Components para optimizar costes
- Vercel (hosting actual) - mantener

Base de Datos:

- Supabase (actual) - extender schema
- Supabase Storage para imágenes
- Vector Store: Supabase pgvector (extensión gratuita)

IA & Procesamiento:

- **Claude 3.5 Sonnet (Anthropic)** - para análisis de imágenes y generación de manuales
- **Claude 3 Haiku** - para el chat en producción (más económico)
- **Vercel AI SDK** - orquestación y streaming
- **LangChain.js** - RAG (Retrieval Augmented Generation)

Búsqueda Web:

- **Brave Search API** (gratuito hasta 2000 queries/mes) o **SerpAPI** (100 búsquedas gratis/mes)

2. ARQUITECTURA DE DATOS

Nuevas Tablas en Supabase o reaprovechamiento de las que hay. Tu decides, pero te pongo el caso de la creación de nuevas tablas → Revisar “fichero1.sql”

Paso 2: Análisis con Claude Vision → “fichero2.js”

Paso 3: Búsqueda Web (si es necesario) → “fichero3.js”

Paso 4: Generación del Manual con Claude → “fichero4.js”

Paso 5: Vectorización para RAG → “fichero5.js”

