RESEARCH BUNDLE

AI SQL Schema Designer

idea-hub · ai-sql-schema-designer

1. The opportunity

นักพัฒนา backend และ full-stack developer ทุกคนต้องออกแบบ database schema ตั้งแต่เริ่มโปรเจกต์ — งานที่กินเวลา 2-8 ชั่วโมงต่อ feature และเป็น bottleneck ก่อน sprint จะเริ่มได้จริง ปัญหาหนักกว่านั้นคือ RLS (Row-Level Security) policy สำหรับ Supabase/PostgreSQL ที่ developer ส่วนใหญ่เขียนผิดหรือ skip ไปก่อน ทำให้เกิด security hole ในภายหลัง ตลาด developer tools กำลังเติบโตบน Supabase ecosystem ที่มีผู้ใช้ active กว่า 1 ล้านโปรเจกต์ในปี 2024 และ indie hacker / startup Thai ที่ build บน Supabase + PostgreSQL เพิ่มขึ้นชัดเจนหลัง Supabase เปิด Bangkok office ช่วงปลายปีที่ผ่านมา Timing ดีเพราะ Text-to-SQL model ฉลาดพอแล้ว แต่ยังไม่มีใครทำ opinionated tool ที่รวม schema + RLS + migration ไว้ในที่เดียว

2. Who would pay

Primary persona: Full-stack developer อายุ 24-35 ปี ทำงานใน startup ไทย/SEA ขนาด 1-15 คน หรือ freelancer ที่รับงาน SaaS/marketplace

- **Job title:** Full-stack Dev, Backend Engineer, Indie Hacker, Technical Co-founder
- **Stack ที่ใช้:** Supabase + Next.js หรือ Remix, บางส่วนใช้ Prisma + PlanetScale
- **สิ่งที่ทำอยู่ตอนนี้:** เปิด ChatGPT แล้ว paste requirement ทีละก้อน 'copy SQL ออกมา 'แก้มือ 'ลืม index 'ทำ RLS ทีหลัง (หรือไม่ทำ
- **Willingness to pay:** ฿299-599/เดือน ถ้า save ได้ 3+ ชั่วโมงต่อสัปดาห์ — developer กลุ่มนี้จ่าย Vercel, Linear, Retool อยู่แล้วไม่ลังเล

Secondary persona: Junior dev ในบริษัท 50-200 คน ที่ต้องส่ง schema ให้ senior review — ใช้ tool นี้เพื่อลด revision round

3. Competitor landscape

ชื่อ	URL	จุดแข็ง	จุดอ่อน
Supabase AI Assistant	supabase.com	Built-in, รู้ context project	Generate SQL ทั่วไป ไม่ได้ opinionated เรื่อง RLS pattern, ไม่ทำ migration file
DBDiagram.io	dbdiagram.io	Visual, ฟรี, ใช้ง่าย	ต้อง draw เอง ไม่มี AI-from-text, ไม่ทำ RLS, export จำกั

RESEARCH BUNDLE

AI SQL Schema Designer

idea-hub · ai-sql-schema-designer

Prisma Data Platform	prisma.io/data-platform	Integration กับ Prisma ORM ดีมาก	เน้น Prisma schema ไม่ใช่ raw SQL/Supabase RLS, pricing สูง ($19+/-mo)
SchemaHero	schemahero.io	Kubernetes-native migration	ซับซ้อนเกินสำหรับ indie dev, ไม่มี AI
SQL Chat (sqlchat.ai)	sqlchat.ai	Chat-based SQL, open source	ตอบ query ไม่ได้ design schema พร้อม RLS + migration ครบ

ช่องว่างที่ชัด: ไม่มีใครทำ Supabase-first schema designer ที่รวม RLS policy generation + versioned migration file ออกมาพร้อมกันใน single flow ตลาดไม่ได้ว่างเพราะไม่มี demand — ว่างเพราะทุกคนคิดว่า "ChatGPT ทำแทนได้" (แต่ UX ของมันไม่ได้ opinionated พอ)

4. Wrapper risk reality check

ใช้ ChatGPT ฟรีได้ไหม? ได้ — แต่ต้องเขียน prompt เอง, ไม่มี structured output, ไม่รู้จัก Supabase RLS pattern ใหม่ๆ, ไม่ generate migration file format ที่ถูกต้อง, และ context หาย ถ้า requirement ยาว

Moat จริงๆ คืออะไร:

- **Opinionated RLS templates:** สร้าง library ของ RLS pattern สำหรับ use case จริง (-multi-tenant SaaS, per-user data isolation, org-level access) ที่ LLM ทั่วไปไม่รู้จักเป็น pattern
- **Migration versioning:** ต่อ schema state ข้าม session ได้ — รู้ว่า migration v3 ต้องต่อจาก v2 อะไร
- **Supabase-specific output:** ออก `supabase/migrations/` format ที่ push ด้วย Supabase CLI ได้ทันที ไม่ต้องแก้มือ
- **Distribution:** ถ้า list ใน Supabase marketplace หรือ integrate กับ Supabase CLI plugin ได้คือ moat เรื่อง distribution

Wrapper risk อยู่ในระดับ **ปานกลาง** — ถ้าไม่มี RLS library + migration state management จะโดน commoditize ใน 12 เดือน

5. Go-to-market angle

10 ลูกค้าแรก หาได้จาก:

1.

RESEARCH BUNDLE

AI SQL Schema Designer

idea-hub · ai-sql-schema-designer

Supabase Discord (discord.supabase.com) — channel `#help` และ `#showcase` เต็มไปด้วย dev ที่ถามเรื่อง RLS ทุกวัน — post case study "เราแก้ปัญหา RLS infinite recursion ด้วย generated policy" แล้วแปะ link

2. **Twitter/X #buildinpublic + #supabase** — founder ควร build in public เลย ทำ thread "วิธี design multi-tenant schema ใน Supabase แบบถูก" แล้วแปะ tool ท้าย thread
3. **Reddit r/Supabase + r/nextjs** — post "I built a tool that generates RLS policies from plain English" — subreddit นี้ชอบ Show HN style post
4. **Thai dev communities** — Facebook group "JavaScript Thailand" (90k+ คน), "React Thailand", Creatorsgarten Discord — ตลาด Thai developer underserved มากใน SaaS tooling
5. **SEO keywords ที่ worth targeting:** "supabase rls policy generator", "postgresql schema design ai", "supabase migration generator", "row level security policy example" — keyword volume ไม่ใหญ่แต่ buyer intent สูงมาก
6. **YouTube SEO:** สร้าง video "Supabase RLS explained + auto-generate" — content gap ชัดมาก ใน Thai/English ก็ยังหายาก
7. **Direct outreach:** หา repo บน GitHub ที่ใช้ Supabase แต่ RLS policy ว่างหรือ `true` — open issue หรือ PR พร้อม generated policy เป็น lead magnet

6. Pricing thesis

Tier	าคา	ขอบเขต
Free	฿0	3 schema/เดือน, ไม่มี migration export
Solo	฿299/เดือน	Unlimited schema, migration files, 5 projects
Team	฿899/เดือน	ทุกอย่าง + team sharing, version history, GitHub export

ทำไม ฿299 ชนะ free alternative: ChatGPT ฟรีแต่ต้องเสีย 30 นาทีต่อ schema + แก้ RLS เองอีก — ถ้า developer คิดค่าเวลาตัวเองที่ ฿500/ชม. save ได้ครั้งเดียวก็คืนทุนแล้ว Annual plan ลด 20% เพื่อ improve cashflow ตั้งแต่เดือนแรก

7. Build complexity

RESEARCH BUNDLE

AI SQL Schema Designer

idea-hub · ai-sql-schema-designer

Time to MVP: 5-7 สัปดาห์สำหรับ solo developer

Core tech stack:

- **Frontend:** Next.js + shadcn/ui (schema editor, visual table diagram)
- **AI Layer:** OpenAI GPT-4o หรือ Claude 3.5 Sonnet + structured output (JSON Schema ' SQL)
- **Schema Visualization:** react-flow สำหรับ ER diagram แบบ interactive
- **Migration Engine:** custom parser ที่ diff schema state v(n) กับ v(n-1) แล้วออก ALTER TABLE statements
- **Auth + Billing:** Supabase Auth (ironic แต่ fitting) + Stripe

Hardest part to ship: Migration diff engine — การคำนวณว่า schema เปลี่ยนยังไงระหว่าง version แล้วออก safe migration ที่ไม่ drop data คือ logic ที่ยากที่สุด ต้อง handle edge case เช่น rename column vs drop+add

What you can fake at first:

- Version 1 ไม่ต้องมี diff engine — แค่ generate full schema + RLS ครั้งแรกให้ดี แล้วให้ user copy-paste เอง
- Visual ER diagram ทำทีหลัง — week 1 แค่มี SQL textarea + download button ก็ขายได้แล้ว
- RLS templates สร้าง 5-10 pattern ที่ cover 80% use case ก่อน (public read, authenticated user, org-based) แทนที่จะ generate สดทุกครั้ง