



Les 11

Vercel AI SDK

Praat met je eigen data — vandaag bouwen we Polderfest 2027



Terugblik

Waar staan we?

- ✓ Supabase geïntegreerd in je app
- ✓ Tabellen + relaties opgezet
- ✓ RLS policies bekeken

Vandaag — geen QuickPoll

We starten een nieuwe demo from scratch.

Next.js + nieuwe Supabase + AI SDK — Polderfest 2027.

Planning

Vandaag — 180 minuten

Welkom + Terugblik	10 min
Theorie: Wat is de Vercel AI SDK?	30 min
LIVE DEMO 1 — Next.js + Supabase	20 min
LIVE DEMO 2 — Seed: 500 records	20 min
Pauze	15 min
LIVE DEMO 3 — AI SDK + chat-route	30 min
LIVE DEMO 4 — Vragen aan onze data	15 min
Lesopdracht + Huiswerk uitleg	20 min
Vragen + Afsluiting	15 min

Concept

Wat is de Vercel AI SDK?

- TypeScript-first, open source
- 40+ providers (OpenAI, Anthropic, Google, Groq...)
- Unified API: zelfde code, elk model
- Streaming out-of-the-box
- React hooks: useChat, useCompletion
- Tool Calling (volgende les)

```
import { generateText } from "ai";
import { openai } from "@ai-sdk/openai";

const { text } = await generateText({
  model: openai("gpt-4o-mini"),
  prompt: "Wat de Polderfest line-up  
samen",
});
```

Wisselen van model = één regel veranderen.

Modellen

Welk model wanneer?

Provider	Model	Use case	Prijs (in/out per 1M)
OpenAI	gpt-4o-mini	Default — snel + goedkoop	\$0.15 / \$0.60
OpenAI	gpt-4o	Multimodaal (vision)	\$2.50 / \$10
OpenAI	gpt-4.1	Reasoning, agents	\$2 / \$8
Anthropic	claude-sonnet-4	Coding, lange context	\$3 / \$15
Google	gemini-2.5-flash	Snel + multimodaal	\$0.075 / \$0.30
Groq	llama-3.3-70b	Ultra-fast inference	\$0.59 / \$0.79

Vuistregel: start met gpt-4o-mini. Onze les vandaag: <2 cent.

Theorie

De 4 kern-functies

Functie	Wat het doet	Wanneer
generateText	Wacht tot klaar — string terug	Korte server-only call
streamText	Stream karakter voor karakter	Chat UI, lange antwoorden
useChat	React hook voor chat UI	Client-side chat
generateObject	Type-safe data via Zod	Database inserts, classificatie

Vandaag gebruiken we:

streamText (server) + useChat (client) — de chat-combo

Volgende les: generateText + tools = Tool Calling

Vandaag bouwen we

Polderfest 2027

Een fictief Nederlands muziekfestival met 500 verzonden bands.

Waarom een fictief festival?

Geen enkele LLM kan dit weten. AI is afhankelijk van onze data. Demo van data + AI.

Tabel: bands

```
name · genre · sub_genre · stage · day · start_time  
origin_city · members · bio · tier · popularity · ticket_impact
```

Vragen die we kunnen stellen:

- Welke bands spelen vrijdagavond op de Main Stage?
- Vat de hip-hop scene samen in 3 zinnen
- Welke acts komen uit Groningen?

LIVE DEMO 1

Next.js scaffold + Supabase koppelen

OpenCode Desktop open op leeg Next.js project. ~20 minuten.

- 1 npx create-next-app** Next.js 15 + TS + Tailwind
- 2 Nieuw Supabase project** Dashboard → New Project
- 3 Schema runnen (SQL Editor)** Tabel bands — leeg
- 4 .env.local** URL + anon + service_role + OpenAI key
- 5 Supabase client** lib/supabase.ts
- 6 Dev server check** npm run dev — alles werkt

Seed script — 500 records procedureel

We lopen door `seed-polderfest.ts` en runnen 'm tegen Supabase. ~20 min.

- 1 Open `seed-polderfest.ts`** Structuur uitleggen
- 2 Bouwstenen tonen** adjectives + nouns → 900 namen mogelijk
- 3 Service role key uitleg** Alleen lokaal, niet client
- 4 `npx tsx scripts/seed-polderfest.ts`** 500 records inserten
- 5 Table Editor verifiëren** 500 bands, allemaal verzonnen
- 6 Quick SQL query** count per genre — verdeling tonen



Pauze

15 minuten



LIVE DEMO 3

AI SDK installeren + chat-route

We bouwen de echte chat. ~30 minuten. Klassikaal, jullie kijken.

1 **npm i ai @ai-sdk/openai zod**

Packages

2 **app/api/chat/route.ts**

Haal bands op → context → streamText

3 **System prompt schrijven**

'Gebruik alleen onze data, verzin niets'

4 **app/chat/page.tsx**

useChat hook + Tailwind UI

5 **Naar /chat browsen**

Eerste vraag testen

6 **Streaming zien werken**

Karakter voor karakter

LIVE DEMO 4

Vragen stellen aan onze data

5 vragen, we lopen ze één voor één door. ~15 min.

1 "Welke bands spelen zaterdag op de Beach Stage?"

Filter

2 "3 headliners met meeste popularity + bios"

Sort + select

3 "Hoeveel jazz fusion acts totaal?"

Aggregate

4 "Wat de electronic-scene samen in 3 zinnen"

Samenvatting

5 "Wie was hoofdact van Polderfest 2025?"

Geen data → eerlijk antwoord

Geen LLM kent Polderfest 2027 — toch werken alle vragen.

Waarom werkt dit?

Data + AI = kracht

Data alleen

- SQL queries
- Filter + sort
- Geen taal
- Geen samenvatting

AI alleen

- Generieke kennis
- Hallucineert vaak
- Geen live data
- Geen privé data

Data + AI

- Interpretatie
- Natuurlijke taal
- Samenvattingen
- Schaalbaar

Een LLM zonder jouw data is een chatbot.

Een LLM mét jouw data is een product.

Lesopdracht

Jouw eigen thema-app

Bouw thuis je eigen versie. Eigen thema, eigen data, eigen vragen.

- [] Bedenk een eigen thema (fictief — geen Spotify, geen restaurants)
- [] Nieuw Next.js project + nieuwe Supabase
- [] Schrijf eigen seed-script (mag AI bij helpen!)
- [] Min 100 records in Supabase
- [] Chat-route + chat-pagina (zelfde flow als Polderfest)
- [] 3 vragen stellen die alleen werken met jouw data

Inspiratie: fictief restaurant-aggregator · scriptie-archief NOVI

museumcollectie · D&D NPCs · cryptid-sightings NL

MOET fictief zijn — anders geen demo-kracht.

Seed aanpassen + uitbreiden + reflectie

Bouwt voort op lesopdracht. Drie onderdelen.

A

Seed-script

Pas Polderfest seed aan voor jouw thema (AI mag helpen). 200+ records.

B

Extra veld

Voeg veld toe dat een nieuwe vraag mogelijk maakt. Test in chat.

C

AI-CHAT.md

Reflectie: thema, 3 goede vragen, 1 vraag die misging + hoe je 'm fixe.

Bonus: deploy op Vercel • loading skeleton • gpt-4o vs mini

Volgende les

Tool Calling — hoe schaalst dit?

Het schaal-probleem

Vandaag: ALLE 500 bands meesturen = ~30k tokens per call
5.000 records? 50.000? Werkt niet meer — te duur, te traag, past niet

De oplossing: Tool Calling

AI besluit zelf welke functie aan te roepen
'Welke bands op vrijdag?' → `searchBands({ day: 'Vrijdag' })`
60 bands terug → AI antwoordt. Schaalbaar, slim, multi-step.

Daarna in deze leerlijn:

- Les 13: Agents + maxSteps (multi-step autonoom)
- Les 14: RAG + embeddings (semantic search op groot corpus)
- Les 15-16: Testing + Deployment • Les 17-18: Eindopdracht + Pitch

Afsluiting

Vragen?

Vandaag gezien:

- Vercel AI SDK basics + 4 kern-functies
- Modellen-landschap + kostenrealiteit
- Next.js + Supabase + AI SDK end-to-end
- Seed script: 500 fictieve records procedureel
- Chat-route + chat-UI met streamText + useChat
- Vragen stellen aan data die geen LLM kan kennen
- Reflectie: data + AI is meer dan beiden apart

Volgende les: Tool Calling

AI besluit zelf welke DB-query te runnen. Schaalbaar tot duizenden records.