



NOVI Hogeschool Utrecht

AI Developer

Lesboek

Alle 18 lessen — opgebouwd in vier delen

Tim Rijkse — schooljaar 2025-2026



Vier delen

Dit lesboek geeft een leesbaar overzicht van het complete AI-Developer curriculum: 18 lessen × 3 uur fysiek, opgebouwd in vier delen. Per les: beschrijving, te behandelen stof, leerdoelen en een verwijzing naar het volledige lesmateriaal in dit repository.

Deel 1

AI Foundations

Les 1-3

Kennismaking met AI, modellen, prompt engineering en de eerste AI-coding tools (ChatGPT, OpenCode, Cursor).

Deel 2

Technical Foundations

Les 4-10

TypeScript en Next.js basics, QuickPoll als rode draad, en Supabase + Auth + Row Level Security tot productie-niveau.

Deel 3

AI Development

Les 11-16

Vercel AI SDK, Tool Calling, Cursor + Vercel deploy, Agents, RAG met embeddings, en Model Context Protocol (MCP).

Deel 4

Production & Advanced

Les 17-18

Observability, evals, security en cost (Les 17). Voice, vision, image gen, local LLMs en edge AI (Les 18).

AI Foundations

Les 1 – 3

Kennismaking met AI, modellen, prompt engineering en de eerste AI-coding tools (ChatGPT, OpenCode, Cursor).

Lessen in dit deel:

L1

Introductie AI & LLMs

L2

AI Code Assistants & OpenCode

L3

Cursor Setup & Basics

Introductie AI & LLMs

Beschrijving

Eerste kennismaking met AI, LLMs en de praktische workflow van schets op papier naar werkende website met ChatGPT en v0.dev.

Te behandelen

- Wat is AI en wat is een Large Language Model (next-token prediction)
- Het AI-landschap: ChatGPT, Claude, Gemini, Grok
- Prompt engineering basics — rol, context, taak, format
- Workflow: schets → ChatGPT → v0.dev → Vercel

Leerdoelen — na deze les kan de student:

- Uitleggen hoe LLM's werken
- Effectieve prompts schrijven met tech stack, kleuren en stijl
- Een schets vertalen naar een werkende landingspagina

Volledig lesmateriaal: [Les01-Introductie-AI/](#)

AI Code Assistants & OpenCode

Beschrijving

Coderen met AI op een fundamenteelere manier: OpenCode als open-source CLI agent. We bouwen een eerste UI met animaties.

Te behandelen

- Waarom OpenCode (open-source CLI agent)
- Tech stack: React, Tailwind, animatie-libraries (GSAP/Lenis)
- Live demo: UI bouwen met OpenCode
- Vergelijking met chatbots: wanneer welk gereedschap?

Leerdoelen — na deze les kan de student:

- OpenCode opzetten en productief gebruiken
- Een UI bouwen via AI-pair-programming
- Beoordelen wanneer welke AI-tool het beste past

Volledig lesmateriaal: [Les02-OpenCode/](#)

Cursor Setup & Basics

Beschrijving

Cursor is dé editor voor AI-development. We zetten hem op, oefenen met de kern-features en lopen door debug-uitdagingen.

Te behandelen

- Cursor installeren en configureren
- Tab autocomplete, Composer, Chat
- Debug-challenges live oplossen
- Cursor rules voor consistente output

Leerdoelen — na deze les kan de student:

- Cursor productief gebruiken voor dagelijks werk
- Composer inzetten voor multi-file edits
- Een debug-flow met AI gestructureerd doorlopen

Volledig lesmateriaal: [Les03-Cursor-Basics/](#)

Technical Foundations

Les 4 – 10

TypeScript en Next.js basics, QuickPoll als rode draad, en Supabase + Auth + Row Level Security tot productie-niveau.

Lessen in dit deel:

L4

TypeScript Fundamentals

L5

Next.js Basics — QuickPoll part 1

L6

Next.js QuickPoll vervolg

L7

Supabase introductie

L8

Van In-Memory naar Supabase

L9

Supabase Auth — eerste login flow

L10

Supabase Auth & RLS verdieping

TypeScript Fundamentals

Beschrijving

TypeScript is de basis van moderne web-development. We behandelen alles wat je nodig hebt om productiecode te schrijven en eindigen met een escape-room vol type-puzzles.

Te behandelen

- Waarom TypeScript boven JavaScript
- Primitives, interfaces, types, generics
- Type narrowing en utility types
- Escape-room met TS-puzzles

Leerdoelen — na deze les kan de student:

- Types schrijven voor productie-code
- Generics gebruiken in echte cases
- TypeScript errors begrijpen en oplossen

Volledig lesmateriaal: [Les04-TypeScript-Fundamentals/](#)

Next.js Basics — QuickPoll part 1

Beschrijving

Eerste echte Next.js-les. We starten de QuickPoll-app die als rode draad door het curriculum loopt.

Te behandelen

- Next.js 16 App Router
- Server vs. client components
- Routing, layouts, loading states
- QuickPoll-app van scratch starten

Leerdoelen — na deze les kan de student:

- Een Next.js-app opzetten met de App Router
- Server- en client-components correct toepassen
- Een eenvoudige interactieve poll-applicatie bouwen

Volledig lesmateriaal: [Les05-NextJS-Basics/](#)

Next.js QuickPoll vervolg

Beschrijving

We bouwen QuickPoll verder uit. Server actions, dynamic routes, form-validatie en een gepolished Shadcn-UI.

Te behandelen

- Server actions in de diepte
- Dynamic routes met [id]
- Form-state, validatie en error-flows
- Tailwind + Shadcn polish

Leerdoelen — na deze les kan de student:

- Server actions schrijven met goede validatie
- Dynamic routes implementeren
- Een gepolished UI bouwen met Shadcn

Volledig lesmateriaal: [Les06-NextJS-QuickPoll-Part2/](#)

Supabase introductie

Beschrijving

Eerste echte database. Supabase: Postgres + auth + storage in één.

Te behandelen

- Wat is Supabase: Postgres, auth, storage in één
- Project aanmaken, tabellen en kolommen ontwerpen
- Supabase Client integreren in Next.js
- Eerste read/write queries

Leerdoelen — na deze les kan de student:

- Een Supabase-project opzetten en koppelen
- Tabellen en relaties ontwerpen
- CRUD-operaties uitvoeren vanuit Next.js

Volledig lesmateriaal: [Les07-Supabase/](#)

Van In-Memory naar Supabase

Beschrijving

QuickPoll gaat écht naar de database. Inclusief real-time votes en foreign keys.

Te behandelen

- Migratie van in-memory data naar Supabase
- Server actions die database-calls doen
- Real-time votes via Supabase Realtime
- Foreign keys en relaties tussen tabellen

Leerdoelen — na deze les kan de student:

- Een bestaande app volledig DB-backed maken
- Real-time updates implementeren
- Database-relaties correct modelleren

Volledig lesmateriaal: [Les08-Van-InMemory-naar-Supabase/](#)

Supabase Auth — eerste login flow

Beschrijving

Eerste echte authenticatie voor QuickPoll. Aan het einde heeft je app signup, login, logout en een werkende Navbar met sessie-status.

Te behandelen

- Authenticatie vs. autorisatie
- Drie methodes: e-mail/password, magic link, OAuth
- Sessie-cookies en @supabase/ssr in Next.js 16
- Middleware-patroon en eerste RLS-policy

Leerdoelen — na deze les kan de student:

- Het verschil tussen authenticatie en autorisatie uitleggen
- Een werkende signup/login/logout-flow bouwen
- Een Next.js middleware schrijven die routes beschermt
- Een eerste RLS-policy formuleren

Volledig lesmateriaal: [Les09-Supabase-Auth/](#)

Supabase Auth & RLS verdieping

Beschrijving

Vervolg op Les 9. We bouwen de auth-flow uit tot productie-kwaliteit en zetten RLS goed in. Aan het einde is QuickPoll multi-user veilig.

Te behandelen

- De drie auth-methodes in detail
- Hoe sessies écht werken: JWT, refresh tokens, cookies
- Row Level Security: using vs. with check
- Beschermd routes via middleware en server components

Leerdoelen — na deze les kan de student:

- Een complete auth-flow opzetten in Next.js + Supabase
- Effectieve RLS-politicijs schrijven met `auth.uid()`
- Het verschil uitleggen tussen auth in code en auth in de database
- Een poll-app maken die multi-user-veilig is

Volledig lesmateriaal: [Les10-Supabase-Auth/](#)

AI Development

Les 11 – 16

Vercel AI SDK, Tool Calling, Cursor + Vercel deploy, Agents, RAG met embeddings, en Model Context Protocol (MCP).

Lessen in dit deel:

- L11** Vercel AI SDK — praat met je data
- L12** Tool Calling
- L13** Externe APIs + Cursor + Vercel deploy
- L14** Agents
- L15** RAG + Embeddings
- L16** Model Context Protocol (MCP)

Vercel AI SDK — praat met je data

Beschrijving

Eerste echte AI-les. We bouwen een Next.js-app from scratch, koppelen Supabase, seeden 500 fictieve bands van Polderfest 2027 en laten studenten vragen stellen aan hun eigen data.

Te behandelen

- Waarom een AI SDK i.p.v. de OpenAI API direct
- De vier kern-functies: generateText, streamText, generateObject, streamObject
- Provider-agnostic werken (OpenAI ↔ Anthropic ↔ Google ↔ Groq)
- Streaming en de UX-impact

Leerdoelen — na deze les kan de student:

- Een Next.js-app koppelen aan Supabase + AI SDK
- Streaming chat implementeren met streamText + useChat
- Een eigen dataset seeden en die met een LLM bevragen
- Het verschil uitleggen tussen direct OpenAI-calls en de AI SDK

Volledig lesmateriaal: [Les11-AI-SDK/](#)

Tool Calling

Beschrijving

Vervolg op Les 11. We lossen het schaalprobleem op: niet alles in context, maar de AI laten kiezen welke tool nodig is. We bouwen 6 tools voor de Polderfest-app.

Te behandelen

- Wat is Tool Calling? Vergelijking met traditionele API-call
- Het schaalprobleem: 30.000 tokens per call werkt niet voor 50.000 records
- `tool({ description, inputSchema, execute })` met Zod
- `stopWhen: stepCountIs(5)` — multi-step reasoning

Leerdoelen — na deze les kan de student:

- Een tool definiëren met `tool()` en Zod-schema
- Multi-step tool calling configureren met `stopWhen`
- Een Supabase-query als tool blootstellen aan de LLM
- Bepalen wanneer tool calling de juiste oplossing is

Volledig lesmateriaal: [Les12-Tool-Calling/](#)

Externe APIs + Cursor + Vercel deploy

Beschrijving

We verlaten localhost. Pokédex met PokéAPI, Cursor Composer + Background Agent voor feature branches, deploy naar Vercel met preview deployments + GitHub Actions CI.

Te behandelen

- Externe APIs in Next.js (server-side fetch, caching)
- Cursor Composer vs. Background Agent
- Vercel preview deployments per branch
- GitHub Actions CI (lint + build per PR)

Leerdoelen — na deze les kan de student:

- Externe APIs aanroepen vanuit een Next.js server-route
- Het verschil tussen Composer en Background Agent uitleggen
- Een feature branch + preview deployment opzetten
- GitHub Actions CI configureren voor een Next.js project

Volledig lesmateriaal: [Les13-Cursor-Vercel-Deploy/](#)

Agents

Beschrijving

Een stap verder dan multi-step tool calling. Agents kunnen 20-50 stappen autonoom uitvoeren, eigen plannen maken en dynamisch bijsturen.

Te behandelen

- Wat is een agent? LLM + loop + tools + stopconditie
- Wanneer wel/niet een agent gebruiken
- ToolLoopAgent, stopWhen, prepareStep
- Dynamisch model + tools per stap wisselen

Leerdoelen — na deze les kan de student:

- Het verschil uitleggen tussen tool calling en een agent
- Een agent opzetten met ToolLoopAgent en custom tools
- Een doordachte stopconditie schrijven
- Modellen dynamisch routen per stap via prepareStep

Volledig lesmateriaal: [Les14-Agents/](#)

RAG + Embeddings

Beschrijving

Hoe laat je een LLM antwoorden op basis van documenten die hij nooit zag? RAG. We bouwen een PDF Q&A-app met embeddings, pgvector en context-injectie.

Te behandelen

- Het probleem: LLM kent jouw documenten niet
- Twee oplossingen: alles meesturen vs. RAG
- Wat zijn embeddings? Vectoren, semantische gelijkenis
- pgvector in Supabase: tabel-setup en index

Leerdoelen — na deze les kan de student:

- Het RAG-pattern uitleggen aan een collega
- pgvector configureren in Supabase
- PDF-content chunken en embedden
- Een werkende vector-search query schrijven

Volledig lesmateriaal: [Les15-RAG-Embeddings/](#)

Model Context Protocol (MCP)

Beschrijving

MCP is de open standaard waarmee AI-clients koppelen aan tools en data. We bouwen een eigen MCP-server voor Polderfest-data en laden hem in Cursor + Claude Desktop.

Te behandelen

- Wat is MCP en waarom bestaat het?
- MCP-architectuur: server, client, transport (stdio, HTTP)
- Tools vs. Resources vs. Prompts
- Bestaande MCP-ecosysteem (filesystem, git, Slack, GitHub)

Leerdoelen — na deze les kan de student:

- MCP uitleggen aan een collega
- Een eigen MCP-server opzetten met de TypeScript SDK
- Een server laden in Cursor en/of Claude Desktop
- Het verschil tussen Tool, Resource en Prompt benoemen

Volledig lesmateriaal: [Les16-MCP/](#)

Production & Advanced

Les 17 – 18

Observability, evals, security en cost (Les 17). Voice, vision, image gen, local LLMs en edge AI (Les 18).

Lessen in dit deel:

L17

AI Production Polish

L18

Advanced AI Toolbox + wrap-up

AI Production Polish

Beschrijving

Van 'het werkt op mijn laptop' naar productie-klaar. Vier pillaren: observability (Langfuse), evals (LLM-as-judge), security (prompt injection + guardrails), cost.

Te behandelen

- De vier pillaren: observability, evals, security, cost
- Waarom logs alleen niet genoeg zijn voor AI
- Evals als unit tests voor LLM-output
- Prompt injection — een echte demo

Leerdoelen — na deze les kan de student:

- De vier productie-pillaren benoemen en uitleggen
- Een AI-app instrumenteren met Langfuse-traces
- Een LLM-as-judge eval-suite opzetten met Vitest
- Cost-per-user meten en model-routing inrichten

Volledig lesmateriaal: [Les17-Production-Polish/](#)

Advanced AI Toolbox + wrap-up

Beschrijving

Laatste les. Vijf korte demo's: voice (Whisper + TTS), vision (GPT-4o), image gen (Flux), local LLMs (Ollama), edge AI (Vercel AI Gateway). Wrap-up + vooruitblik eindopdracht.

Te behandelen

- Het AI-landschap eind 2026: providers en modi
- Per feature: welke modaliteit past?
- Open-source vs. cloud: privacy, cost, latency
- Resources en community om bij te blijven

Leerdoelen — na deze les kan de student:

- Het AI-landschap eind 2026 schetsen
- Per feature de juiste modaliteit kiezen
- Een voice- of vision-feature implementeren
- Een lokaal model draaien via Ollama
- Vercel AI Gateway gebruiken voor automatic fallback

Volledig lesmateriaal: [Les18-AI-Toolbox-Next/](#)

Eindopdracht

Wat

De student bouwt zelfstandig — thuis, na de 18 lessen — een eigen AI-app die de volledige stack uit de leerlijn integreert. De eindopdracht is geen in-class lesonderdeel.

Verplichte componenten

- Next.js 16 + TypeScript + Tailwind / Shadcn
- Supabase met authenticatie en Row Level Security
- Vercel AI SDK met Tool Calling
- Minimaal één externe API
- Deployed op Vercel met preview deployments per branch

Pitch: 8-12 minuten demonstratie na inleveren.

Volledige eisen, beoordelingscriteria en deadlines:

[Eindopdracht-AI-Developer.md](#)



Colofon

AI Developer

Lesboek — versie juni 2026

Opleiding	NOVI Hogeschool Utrecht
Vak	AI-Assisted Development
Docent	Tim Rijkse
Schooljaar	2025–2026
Omvang	18 lessen × 3 uur fysiek