



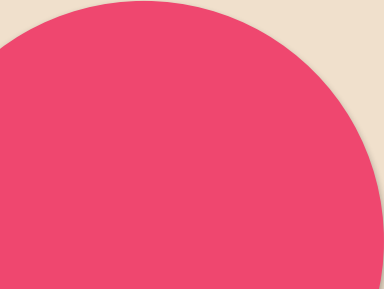
NOVI Hogeschool Utrecht

AI Developer

Curriculum-overzicht

18 lessen × 3 uur fysiek

Tim Rijkse — schooljaar 2025-2026



Vier delen

Deel 1

AI Foundations

Les 1-3

Deel 2

Technical Foundations

Les 4-10

Deel 3

AI Development

Les 11-16

Deel 4

Production & Advanced

Les 17-18

Overzicht

Alle 18 lessen

L1

Introductie AI & LLMs

L2

AI Code Assistants & OpenCode

L3

Cursor Setup & Basics

L4

TypeScript Fundamentals

L5

Next.js Basics — QuickPoll part 1

L6

Next.js QuickPoll part 2

L7

Supabase introductie

L8

Van In-Memory naar Supabase

L9

Supabase Auth — eerste login flow

L10

Supabase Auth & RLS verdieping

L11

Vercel AI SDK — praat met je data

L12

Tool Calling

L13

Externe APIs + Cursor + Vercel

L14

Agents

L15

RAG + Embeddings

L16

Model Context Protocol (MCP)

L17

AI Production Polish

L18

Advanced AI Toolbox + wrap-up

Introductie AI & LLMs

Van ChatGPT tot v0.dev

Te behandelen

- Wat is AI, wat is een LLM (next-token prediction)
- Het AI-landschap (ChatGPT, Claude, Gemini, Grok)
- Prompt engineering basics
- Workflow: schets → ChatGPT → v0.dev → Vercel

Leerdoelen — na deze les kan de student:

- LLM's uitleggen
- Effectieve prompts schrijven
- Een schets vertalen naar een werkende landingspagina

AI Code Assistants & OpenCode

OpenCode, slim werken, eerste UI's

Te behandelen

- Waarom OpenCode (open-source CLI agent)
- Tech stack: React, Tailwind, animaties (GSAP/Lenis)
- Live demo: UI bouwen met OpenCode
- Vergelijking met chatbots

Leerdoelen — na deze les kan de student:

- OpenCode opzetten en gebruiken
- Een UI bouwen met AI-pair-programming
- Beoordelen wanneer welke AI-tool past

Cursor Setup & Basics

De IDE voor AI-development

Te behandelen

- Cursor installeren + configureren
- Tab autocomplete, Composer, Chat
- Debug-challenges live oplossen
- Cursor rules

Leerdoelen — na deze les kan de student:

- Cursor productief gebruiken
- Composer inzetten voor multi-file edits
- Debug-flow met AI structureren

TypeScript Fundamentals

Types, generics, escape-room

Te behandelen

- Waarom TypeScript
- Primitives, interfaces, types, generics
- Type narrowing + 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

Next.js Basics — QuickPoll part 1

App Router, server components, eerste poll-app

Te behandelen

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

Leerdoelen — na deze les kan de student:

- Een Next.js-app from scratch opzetten
- Server components correct toepassen
- Een eenvoudige interactieve poll bouwen

Next.js QuickPoll part 2

Forms, server actions, dynamic routes

Te behandelen

- Server actions diep
- Dynamic routes [id]
- Form-state, validatie, errors
- 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

Supabase introductie

Postgres, dashboard, eerste tabel

Te behandelen

- Wat is Supabase (Postgres + auth + storage)
- Project aanmaken, tabellen, kolommen
- Supabase Client in Next.js
- Eerste read/write queries

Leerdoelen — na deze les kan de student:

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

Van In-Memory naar Supabase

QuickPoll écht naar database

Te behandelen

- Migratie: in-memory naar Supabase
- Server actions die DB-calls doen
- Real-time votes met Supabase Realtime
- Foreign keys en relaties

Leerdoelen — na deze les kan de student:

- QuickPoll volledig DB-backed maken
- Real-time updates implementeren
- Database-relaties correct modelleren

Supabase Auth — eerste login flow

signUp, signIn, signOut, Navbar, RLS intro

Te behandelen

- Authenticatie vs. autorisatie
- Email/password, magic link, OAuth
- Sessie-cookies + @supabase/ssr
- Middleware + eerste RLS-policy

Leerdoelen — na deze les kan de student:

- Auth vs. autorisatie uitleggen
- Een complete login-flow bouwen
- Een eerste RLS-policy schrijven

Supabase Auth & RLS verdieping

Multi-user veiligheid op database-niveau

Te behandelen

- Drie auth-methodes in detail
- JWT, refresh tokens, cookies
- RLS: using vs. with check
- Beschermd routes via twee patronen

Leerdoelen — na deze les kan de student:

- Effectieve RLS-policijs schrijven
- Sessies veilig beheren
- Multi-user veilige apps bouwen

Vercel AI SDK — praat met je data

Polderfest 2027: 500 bands in Supabase

Te behandelen

- Vier kern-functies: generateText, streamText, generateObject, streamObject
- Provider-agnostic werken
- Live demo: Next.js + Supabase + AI SDK
- Eerste chat-route met streaming

Leerdoelen — na deze les kan de student:

- Een AI-app opzetten met de Vercel AI SDK
- Streaming chat implementeren
- Een eigen dataset bevragen met een LLM

Tool Calling

Laat AI zelf functies kiezen

Te behandelen

- Het schaalprobleem: niet alles in context
- `tool({ description, inputSchema, execute })`
- `stopWhen` + multi-step
- 6 tools voor de Polderfest-app

Leerdoelen — na deze les kan de student:

- Tools definiëren met Zod-schemas
- Multi-step tool calling configureren
- Bepalen wanneer tool calling past

Externe APIs + Cursor + Vercel

Van localhost naar de wereld

Te behandelen

- Externe APIs in Next.js
- Cursor Composer vs. Background Agent
- Vercel preview deployments
- GitHub Actions CI

Leerdoelen — na deze les kan de student:

- Externe APIs integreren
- Cursor agents productief inzetten
- Een Vercel-deploy met CI opzetten

Agents

LLM in een loop met tools

Te behandelen

- Wat is een agent (LLM + loop + tools + stopconditie)
- ToolLoopAgent, stopWhen, prepareStep
- Dynamisch model-routing per stap
- Research-agent from scratch

Leerdoelen — na deze les kan de student:

- Een agent bouwen met de AI SDK
- Slimme stop-condities schrijven
- Modellen dynamisch routen

RAG + Embeddings

Praat met je documenten

Te behandelen

- Het RAG-pattern: chunk → embed → store → query
- pgvector in Supabase
- Vector-search query schrijven
- PDF Q&A app from scratch

Leerdoelen — na deze les kan de student:

- Het RAG-pattern uitleggen
- pgvector configureren en gebruiken
- Een werkende RAG-app bouwen

Model Context Protocol (MCP)

Eén protocol, alle AI-clients

Te behandelen

- Wat is MCP en waarom
- Tools, Resources, Prompts
- MCP-server in Cursor + Claude Desktop
- Eigen Polderfest MCP-server

Leerdoelen — na deze les kan de student:

- Een MCP-server bouwen met de TypeScript SDK
- MCP integreren in Cursor en Claude Desktop
- Bepalen wanneer MCP vs. inline tools past

AI Production Polish

Observability, evals, security, cost

Te behandelen

- Vier pillaren van productie-AI
- Langfuse: tracing + logging
- LLM-as-judge evals (Vitest)
- Prompt injection + cost monitoring

Leerdoelen — na deze les kan de student:

- AI-apps instrumenteren met Langfuse
- Een eval-suite opzetten
- Security en cost beheren

Advanced AI Toolbox + wrap-up

Voice, vision, image, local, edge

Te behandelen

- Voice (Whisper + TTS)
- Vision (GPT-4o)
- Image generation (Flux via Fal.ai)
- Local LLMs (Ollama) + edge AI (Vercel Gateway)

Leerdoelen — na deze les kan de student:

- Per feature de juiste modaliteit kiezen
- Voice-, vision- en image-features bouwen
- Lokale modellen en edge-AI inzetten

Afsluiting

Eindopdracht

De student bouwt een eigen AI-app — thuis, na de 18 lessen.

Verplichte componenten:

- Next.js + TypeScript + Tailwind / Shadcn
- Supabase met auth en RLS
- Vercel AI SDK met Tool Calling
- Minimaal één externe API
- Deployed op Vercel met preview deployments

Pitch: 8–12 minuten demonstratie na inleveren.



Colofon

AI Developer

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

