Klas Olsson

Fullstack- och AI-utvecklare · Göteborg

klasolsson81@gmail.com · linkedin.com/in/klasolsson81 · github.com/klasolsson81

Profil

Jag bygger produkter från början till slut: datamodellen, API:et, gränssnittet och servern de går på.

Innan jag började med mjukvara arbetade jag mer än 20 år i fordonsindustrin, som operatör, teamledare och projektledare. Där lärde jag mig att bry mig om processer och om sådant som ska fortsätta gå.

Jag läser till systemutvecklare .NET på NBI/Handelsakademin i Göteborg (2025–2027). Min första LIA gjorde jag på Yobber, som fullstackutvecklare och teknisk ledare, och i februari 2027 går jag tillbaka dit för den andra.

Jag bygger in språkmodeller i produkter med skydd för personuppgifter, och jag utvecklar med AI-agenter i Claude Code och Codex.

Öppen för utvecklarroller och konsultuppdrag, på plats runt Göteborg eller på distans i hela Sverige.

Erfarenhet

Fullstackutvecklare och teknisk ledare (LIA) — Yobber **augusti–november 2026**

- Ledde den tekniska planeringen med vd och bröt ned arbetet i 200+ ärenden i GitHub.
- Fördelade uppgifter till praktikanterna, stöttade dem en och en och granskade deras PR:er.
- Utvecklade rekryteringsplattformen i React, TypeScript och Supabase, med 100+ egna sammanslagna PR:er.
- Byggde AI-analys av ansökningar med Gemini och funktionsanrop, i två flöden.
- Ansvarade för arkitektur, säkerhet och GDPR i modulen för personlighetstest. Skyddade AI-omdömet mot diskriminering, och dokumenterade det i en konsekvensbedömning (DPIA).
- Byggde en AI-funktion som ändrar en annons efter en instruktion, och som sparar ändringen först när en människa har godkänt den.
- Lade till 300+ testfiler i Vitest och 100+ databasmigreringar i Supabase.

Grundare och utvecklare — Jobbliggaren (eget projekt) **2026–nu**

- Byggde Jobbliggaren.se ensam i .NET, C#, Next.js och PostgreSQL, med 1 000+ egna sammanslagna PR:er.
- Drev utvecklingen med AI-agenter i Claude Code och Codex, i upp till 4 parallella sessioner.
- Skapade 10+ specialiserade AI-agenter för arkitektur, säkerhet, test och kodgranskning.
- Designade backend med Clean Architecture, DDD och CQRS, styrd av 100+ arkitekturbeslut (ADR).
- Byggde inloggning utan lösenord, OAuth med Google, LinkedIn och GitHub, och kryptering med AES-256.
- Driftsatte hela stacken på en tom Linux-server med Docker, Caddy, Redis och CI/CD, i drift dygnet runt.
- Säkrade kvaliteten med 30 000+ automatiserade tester och arkitekturtester som stoppar bygget.
- Utvecklade en regelbaserad CV-granskare, medvetet utan språkmodell för spårbarhetens skull, med 20+ regler ur en mall med 40+ kriterier.

Grundare och utvecklare — KalasKoll (eget projekt) **2026–nu**

- Lanserade kalaskoll.se, digitala inbjudningar till barnkalas, som har vuxit till 80+ medlemmar utan marknadsföring.
- Byggde två AI-funktioner: en supportchatt med Claude, och inbjudningskort som ritas med Flux och DALL-E.
- Fick KalasKoll att växa genom SEO och kundomdömen, efter ett lanseringsinlägg med 60 000+ visningar på LinkedIn.
- Byggde svar via QR-kod, krypterade allergiuppgifter (AES-256) och betalning med Stripe i Next.js.

Operatör, tidigare teamledare och projektledare — Plasman **2005–2026**

- Ledde team och förbättringsprojekt som teamledare, projektledare och handledare.

Operatör, inhyrd via Lernia — Plastal (i dag Plasman) **2002–2005**

Utbildning

Systemutvecklare .NET — NBI/Handelsakademin, Göteborg **2025–2027**

Leder till yrkeshögskoleexamen. Avancerad C#, Clean Architecture, databasdesign, Azure, agil metodik och fullstack med .NET och React.

Högskolestudier i elkraft och värmeteknik — Högskolan i Borås **2000–2002**

44 högskolepoäng inom högskoleingenjörsprogrammet. Avslutade kurser i matematik, datorlära och ellära.

Naturvetenskapsprogrammet, teknisk inriktning — Ale gymnasium **1997–2000**

Gymnasieexamen

Kompetenser

AI och språkmodeller: Claude (Anthropic API), OpenAI API, Gemini, funktionsanrop, promptdesign, Vercel AI SDK, ElevenLabs (tal och röstklon), Replicate och Flux (AI-bilder)

AI-driven utveckling: Claude Code, Codex, egna AI-agenter och skills, parallella agentsessioner

AI-säkerhet: skydd mot promptinjektion, GDPR och konsekvensbedömning för AI, mänskligt godkännande

Programspråk: C#, TypeScript, JavaScript, SQL, Python, Bash, PowerShell

Backend: .NET, ASP.NET Core, ASP.NET Core Web API, ASP.NET Core Identity, Minimal API, EF Core, ADO.NET, LINQ, Mediator (CQRS), FluentValidation, AutoMapper, SignalR, Hangfire, RabbitMQ, Serilog och Seq, OpenAPI (Swagger, Scalar), Stripe (betalning och webhooks)

Frontend: React, Next.js, HTML, CSS, Tailwind CSS, shadcn/ui, React Hook Form, Zod, Zustand, Framer Motion, Vite, Astro, PWA, i18n (next-intl, i18next), tillgänglighet (WCAG 2.2 AA), SEO

3D och spel: three.js, React Three Fiber, GLSL, Blender (med Python), glTF, Phaser 3

Databaser: PostgreSQL, SQL Server, MySQL, SQLite, Supabase, Redis, T-SQL (vyer, triggers, lagrade procedurer), PL/pgSQL, ER-modellering och normalisering, fulltextsökning i PostgreSQL, databasmigreringar, radnivåsäkerhet (RLS)

Arkitektur och test: Clean Architecture, DDD, CQRS, SOLID, designmönster, Repository-mönstret, UML, kravanalys och användningsfall, TDD, xUnit, NUnit, MSTest, Moq, FluentAssertions, NSubstitute, Testcontainers, Vitest, React Testing Library, Playwright, arkitekturtester (NetArchTest), lasttester (NBomber), Lighthouse CI

Säkerhet: OAuth 2.0, lösenordsfri inloggning, JWT, rollbaserad behörighet, kryptering (AES-256), BCrypt, OWASP Top 10 och ASVS, CVSS, CSP och HSTS, leveranskedjans säkerhet (Trivy, Cosign, CodeQL, Dependabot)

Drift och DevOps: Docker, Docker Compose, Caddy, Linux-server, systemd, GitHub Actions, CI/CD, Terraform, AWS (ECS Fargate, RDS, ElastiCache), Vercel, Netlify, Azure, Upstash

Arbetssätt: Scrum, Kanban, GitHub Flow, kodgranskning, ärendeplanering i GitHub

Projekt

klasolsson.se

Astro, TypeScript, three.js, Blender, Claude (Anthropic API), ElevenLabs, Vercel, Upstash Redis

Min portfolio: en hall i 3D i webbläsaren, där en AI-kopia av mig svarar på frågor med min röst.

Console Detective

C#, .NET, React, TypeScript, OpenAI API

Mitt första egna projekt: ett noir-detektivspel med AI-genererade fall. Det byggdes först som konsolapp och sedan som React-klient mot ett .NET-API.

Sky High Adventures

React, Phaser 3

Ett endless runner-spel i webbläsaren, byggt i React och Phaser 3.

Språk

Svenska: modersmål

Engelska: flytande i tal och skrift