

Igor Tarasenko — AI Architect CV

Igor Tarasenko

AI Architect / Agentic Engineering Lead · Amsterdam, NL

Email: igor@tarasenko.dev · Website: <https://tarasenko.dev> · GitHub: <https://github.com/Saik0s> · X: <https://x.com/saik0s>

Summary

AI architect and senior software engineer with 16 years building production software across mobile, platform, payments, developer tooling, and applied AI. Ex-Uber. Currently lead iOS engineer and internal AI enablement lead at DNA.inc, helping engineering teams adopt agentic development workflows that hold up against real codebases.

I work at the intersection of iOS, LLM systems, agentic tooling, and production reliability: evals, guardrails, MCP tools, on-device inference, AI coding workflows, and pragmatic architecture that turns prototypes into systems teams can maintain.

Architecture strengths

- **Production AI systems:** LLM integration, structured outputs, eval loops, guardrails, cost/latency trade-offs, fallback design.
 - **Agentic engineering:** Claude Code/Cursor workflows, MCP tooling, agent-readable repo conventions, build/test/preview feedback loops.
 - **Mobile AI:** Swift, SwiftUI, TCA, UIKit, on-device Whisper, App Store constraints, privacy-first architecture.
 - **Developer experience:** internal frameworks, automation, build systems, codebase legibility, onboarding, team enablement.
 - **Applied generative media:** ComfyUI pipelines, Stable Diffusion / diffusion-video workflows, local model operations.
 - **Scale judgment:** Uber platform/payments background; systems where reliability, cross-team ownership, and failure modes matter.
-

Experience

DNA.inc — Lead iOS Engineer & Internal AI Enablement

Lead iOS engineering while also acting as the practical AI adoption lead for engineering workflows.

- Lead work on production iOS systems in a large enterprise SaaS environment.
- Run internal AI enablement sessions for engineers: agentic coding workflows, practical model use, review loops, and guardrails.
- Design workflows for safe AI-assisted development: context files, deterministic checks, repo conventions, build/test feedback.
- Evaluate where LLMs are useful, where they fail, and how to structure the work so senior engineers retain control.

Uber — Senior Software Engineer

Worked on platform and payments systems at Uber scale.

- Built internal server-driven UI infrastructure and developer-facing platform systems.
- Worked on payments systems where reliability and cross-team ownership mattered at global scale.
- Operated in large-codebase environments with build-system complexity, architecture review, and multi-team coordination.

Independent / Open Source — AI products, tools, and consulting

Built production tools and open-source systems around iOS, AI, developer tooling, and automation.

- Built **WhisperBoard**, an on-device iOS transcription app: 50k+ downloads, 4.8 App Store rating, 1,035 GitHub stars.
- Built **mcp-browser-use**, a Python MCP server for browser automation by AI agents: 938 GitHub stars.
- Built **VibeSwitch**, an AI keyboard for rewriting text with sub-cent LLM call economics.
- Built and operated local ComfyUI / diffusion pipelines for image and video generation, including Stable Diffusion-family workflows.
- Mentor engineers through Codementor on iOS, AI integration, architecture, and practical debugging.

Selected proof

WhisperBoard

Open-source iOS app for on-device voice transcription.

- 50k+ downloads.
- 4.8 App Store rating.
- 1,035 GitHub stars.
- Demonstrates mobile inference, privacy-first architecture, model packaging, App Store delivery, and real user traction.

mcp-browser-use

Open-source MCP server for browser automation by AI agents.

- 938 GitHub stars.

- Demonstrates early MCP ecosystem work, agent-facing tool design, browser automation, and Python infrastructure.

Agentic iOS development workflow

Six-plus months of agent-driven iOS development before Apple's native Xcode agent support became mainstream.

- Built preview-feedback tooling because agent loops fail without fast visual verification.
- Structured repositories around explicit agent-readable conventions, build scripts, and review gates.
- Practical thesis: the bottleneck is usually feedback-loop design, not prompt wording.

ComfyUI / Stable Diffusion production workflows

Hands-on work with local generative pipelines, including Stable Diffusion-family workflows and video generation.

- Model management, GPU workflow debugging, failure handling, and reproducible local pipelines.
- Focus on turning experimental workflows into repeatable systems rather than demo notebooks.

Technology

Languages: Swift, Python, TypeScript, JavaScript, PHP

iOS: SwiftUI, UIKit, TCA, Tuist, App Store delivery, on-device ML

AI/LLM: Anthropic, OpenAI, Claude Code, Cursor, MCP, structured output, evals, guardrails

Generative media: ComfyUI, Stable Diffusion-family models, image/video generation pipelines

Backend/data: FastAPI, Postgres/Supabase, SQLite, vector/RAG trade-offs

Infra: Hetzner, Coolify, Vercel, CI/CD, build automation

Practices: agentic development, code review systems, test harnesses, cost engineering, developer enablement

Education / background

Started programming at 16. Originally from Ukraine; based in Amsterdam. Long-running focus on building tools that compress the distance between idea, code, and shipped product.

Links

- Website: <https://tarasenko.dev>
- GitHub: <https://github.com/Saik0s>
- X: <https://x.com/saik0s>
- Codementor: <https://www.codementor.io/@saik0s>