Lukas Janda

Architect & full-stack engineer · 18 years · Remote only · Czech Republic (CET) · English C1
hey@lukasjanda.com · lukasjanda.com · github.com/lukasjandacom · writing

Architect and full-stack engineer, **18 years** shipping production software. Ten of those running my own software company, which I grew from the first paying client to a **~15-engineer team**; the last eight architecting and owning a multi-service platform — 11 backend services, 6 front-end apps, AI document pipelines, computer vision and integrations.

The thing worth knowing: I have run delivery at **both scales** — a ~15-person team, and a 2–3 person team hitting the same scope with an AI-accelerated workflow. Measured, not projected. I agree the workflow with the people who will use the thing, then get a prototype into real use fast and tune against reality.

// experience

2018 – now **Architect & Product Owner** a multi-service platform at a product company

- Own the architecture and the product: **11 backend services, 6 front-end applications**, multi-tenant across **27 locations in 9 countries**. Every feature began as a flow I sketched and a call I made about how it should work.
- **Offline-first field apps** — a durable local queue, ordered replay, conflicts resolved against server state.
- **AI document pipelines** where each step retries alone, so one failed extraction does not cost the run.
- **Integrations** that keep separate systems in agreement across APIs neither side controls. Test suites in the four figures.

2008 – 2018 **Founder / Engineering Lead** own software company

- Grew it from the first paying client to a **~15-engineer team**, staying technical throughout: architecture, hiring, delivery and the client relationships that funded all three.
- **100+ projects delivered.** Most under NDA; I do not publish client names, and would extend the same discretion to an employer.
- The part that transfers: deciding what *not* to build, and being accountable for the estimate.

// models — trained, shaped or put to work

Part and wheel segmentation	YOLO segmentation of 23 part classes, in production	co-trained	platform
Image tagging	part, side and travel direction per image	co-trained	platform
Camera-position-aware models	a model per camera layout, plus rules for what each sees	co-architected	platform
Capture guidance	right level, distance and angle, images collected automatically	co-architected	platform
Face anonymisation	detector plus a false-positive classifier on reviewed crops	co-architected	platform
Part detection in a capture app	YOLO with a vision-LLM fallback when unsure	prototype	handed over
Rubbish and clean-up photos	MobileNetV3 fine-tuned on my labels, clean or not 93 %	trained	Flek
Privacy filters in the phone	faces pixelated, explicit photos held, before upload	integrated	Flek

// work I own outright — no NDA, live, open to read

Flek flekit.com · source
A civic map of small neglected places, with no accounts at all. Offline-first, over 1,300 tests.

A lead-discovery engine writing
Nine sources swept nightly, scored by an LLM in two lanes, one ranked digest. Eligibility moved out of the prompt and into code after the model misread a hard constraint four times. Runs unattended, 200+ tests.

// stack

languages	Python, Kotlin, JavaScript, TypeScript
frontend	Vue 3, React, MapLibre, PWA
backend	Node, Python/FastAPI, Kotlin/Spring
data	PostgreSQL, SQLite/D1, Supabase
ai	Claude & OpenAI APIs, RAG and extraction pipelines, YOLO and other vision models
platform	Cloudflare Workers/Pages/KV/R2, serverless-first
delivered	international multi-tenant platform · PWA for phones · machine learning in production
beyond code	product ownership · software architecture · project management · tech support · hands-on UI testing

// what I am looking for

Contract, fixed-scope or fractional-lead work, fully remote, in product engineering; I invoice as a registered Czech sole trader. A fully remote full-time role is also of interest.

Not infrastructure ownership: no DevOps, no SRE, no on-call, no platform engineering.