

// HANDS-ON SOFTWARE ARCHITECT

Troels Lenda Strand

I design systems, write code, and remove friction.

```
$ echo $HEART > my heart runs Linux; everything else  
is in compatibility mode.
```

01 - WHO I AM

Who I Am

Senior software developer with **20+ years** in the industry. I thrive in the space between architecture and implementation — close enough to whiteboards to shape direction, close enough to the code to know what it costs.

Still hands-on, and not particularly interested in stopping. I care about sustainable systems, clean abstractions, developer experience, and code that someone — possibly me, possibly a stranger — can still reason about in three years.

When I encounter the same friction twice, I try to turn it into a tool instead of just complaining about it.



How I Think

A few questions I find myself asking, often:

” Shouldn’t we streamline this ?”

” Can this become a shared capability ?”

” Can we generalize this with a small compromise ?”

” Why are we rebuilding what Elasticsearch already solves ?”

” Can tooling solve this permanently ?”

Good architecture, to me, is mostly about reducing friction and respecting the strengths of the platforms you already stand on. Don’t out-clever the database. Don’t reinvent the search engine. Make the easy thing the right thing.

What I Bring

01 Architecture & Systems

API design, reusable capabilities, search architecture, event-driven systems, pragmatic abstractions, maintainability.

02 Frontend

Angular, TypeScript, JavaScript, CSS, SCSS, HTML, content-driven platforms.

03 Backend & Infrastructure

Java (Spring Boot & Quarkus), Node.js, Firebase/GCP, Elasticsearch, OpenShift, Docker & k8s, CI/CD, Linux, nginx, databases.

04 Developer Experience

Internal tooling, automation, local development workflows, repository navigation, onboarding simplification.

05 Platforms & CMS

Drupal, Umbraco, TYPO3, content management.

Selected Work

Bankdata

2021-PRESENT · SELF-SERVICE INVESTMENT PLATFORM

- PROBLEM** Regulated banking applications with long release cycles, high reliability expectations, and product surfaces that still need to evolve.
- CONTRIBUTION** Angular frontend architecture alongside Java services, Elasticsearch, OpenShift, and CI/CD. Contributed to design-system work and used backend-for-frontend patterns to let selected presentation behavior be driven more flexibly, almost CMS-like, when a full frontend release was too heavy.
- FOCUS** Maintainable financial systems, delivery flexibility, and pragmatic architecture under real constraints.

DGI IT

2019-2021 · INTERNAL APPLICATIONS & DEVOPS

- PROBLEM** Internal tools built on top of a core system where some design problems were deeply embedded and could not be cleanly replaced inside the available constraints.
- CONTRIBUTION** Worked as an architect-minded frontend developer across Angular, Umbraco, and DevOps. Identified structural design issues, pushed repeatedly for better direction, and found a workable compromise when the ideal fix was too entangled with the core system.
- FOCUS** Practical architecture in legacy constraints: reducing local complexity while keeping delivery moving.

Makeable

2017-2019 · BACKEND / SERVERLESS ARCHITECTURE

- PROBLEM** Product ideas that needed realtime behavior, payments, mobile experiences, and physical device integration without a heavy infrastructure footprint.
- CONTRIBUTION** Built Firebase-based product architecture for [Dripdrop.io](https://dripdrop.io) and Wedbox. For Dripdrop, combined focused Node.js cloud functions, Realtime Database state, WebSockets, Stripe payments, [Particle.io](https://particle.io), LTE-connected hardware, and hands-on device work into an umbrella-sharing experience. I also wrote about my findings on [Firebase Cloud Functions trigger latency](#).
- FOCUS** Serverless architecture that matched the business case: small responsibilities, event-driven flows, and a tight bridge between app, cloud, and hardware.

Reload

2010-2017 · SENIOR FULLSTACK & FRONTEND LEAD

- PROBLEM** Large CMS and web platforms with long lifespans and many stakeholders.
- CONTRIBUTION** Frontend lead and senior fullstack on dds.dk, spejder.dk, ida.dk, scleroseforeningen.dk, gocook.dk and more, with additional work on dr.dk and Ørsted CRM.
- FOCUS** Sustainable platforms editors and developers can both live with.

Developer Experience

“When I encounter repeated friction, I usually build a tool and share it.”

In professional codebases, repeated friction is usually a design signal. I build repository navigation helpers, automation scripts, local development improvements, and internal tooling when the same small obstacle keeps slowing people down.

None of it is glamorous. All of it compounds: less waiting, less guessing, fewer manual steps, and more attention left for the work that actually needs judgment.

AI & Engineering

I use AI-assisted development to accelerate prototyping, scaffolding, and first-pass review work — while maintaining full ownership of architecture, code quality, and production decisions.

Used well, AI reduces review overhead and helps get from idea to working shape faster. The taste, and the responsibility, remain mine.

troelslenda@gmail.com / github.com/troelslenda / linkedin.com/in/troelslenda