Stéphan Zych

Lead Developer · Application & Web Architect · Technical Lead

Brussels, Belgium

stephan@zych.be stephan.zych.be linkedin.com/in/stephanzych github.com/monkeymonk

ABOUT ME

Hands-on lead developer and application architect, 15+ years across agency, founder and international consulting work. I design web and application architecture in the PHP and JavaScript/TypeScript ecosystems, own the technical decisions around it (stack choices, project structure, standards, code review), and keep implementing alongside the team.

I also focus on how software gets built and shipped: reusable project foundations, containerised environments, CI/CD, automated deployments and disciplined AI-assisted engineering.

CORE EXPERTISE

Architecture & technical leadership

Application & web architecture · Technical direction · Stack & tooling decisions · API / backend / frontend boundaries · Technical standards & code review · Legacy modernisation

Platform, delivery & developer experience

Docker environments · CI/CD & automated deployments · Reusable project foundations · Engineering standards · Tooling & automation

Application engineering

PHP platforms (Laravel, WordPress) · JavaScript / TypeScript applications · Headless & API integration · Accessibility & semantic markup · Web performance

AI-assisted engineering

Coding agents in daily development · Shared agent configuration (skills, MCP, rules) · AI-assisted refactoring & legacy analysis · Sandboxed agent workflows · Local LLM tooling

WORK EXPERIENCE

Lead Developer — CBTW · via STEPHANZYCH

Brussels, Belgium · 2023 – Present

Hands-on lead developer in the technical team that came out of the Cherry Pulp integration, currently four to five engineers. I develop day to day while leading technical direction and the modernisation of how the team builds and ships.

Architecture & technical direction

- Own architecture and technical decisions across concurrent projects: stack choices, project structure, and the boundaries between backend, API and frontend
- Define technical standards, development conventions and code review practice, and drive their adoption across teams
- Maintain and evolve the internal tooling and project boilerplates originally built at Cherry Pulp and still in use within CBTW
- Support developers through review and technical guidance while remaining in the codebase

- Work across distributed international teams, including enterprise environments using Microsoft internal tooling and AWS

Platform & delivery modernisation (in progress)

- Introduced a shared Docker local development environment (Traefik, PHP, MariaDB, phpMyAdmin) as the common baseline for projects
- Building GitLab CI pipelines and secured containerised deployment on AWS Lightsail, replacing manual and FTP-based deployment and moving off Laravel Forge
- Adding observability with Grafana, Prometheus and Loki
- Rolled out project by project alongside ongoing client delivery: legacy workflows and the new pipeline currently run side by side

---

Co-Founder / CTO / Head of Frontend — Cherry Pulp

Brussels, Belgium · 2013 – 2024 · acquired by CBTW in 2023, transition completed in 2024

Co-founded Cherry Pulp and helped grow it into a digital agency of around 20 people building custom websites, digital platforms and web applications. My role ran from frontend leadership into technical ownership across the company, hands-on throughout.

Scale & impact

- Delivered 200+ web platforms and digital products
- Helped grow the company from 4 to 20 people
- Supported and mentored development teams, up to 10–12 developers at peak

Technical ownership & delivery

- Designed a modular web delivery approach around Docker, WordPress with Bedrock, structured conventions and custom Forge/Lightsail deployment workflows, making setup faster and delivery more consistent across teams
- Defined technical direction and project architecture across client work: stack selection, project structure, and the boundaries between WordPress or Laravel backends, APIs and JavaScript frontends
- Owned frontend technical direction and the delivery of short-cycle web projects: WordPress from small to medium complexity, Laravel applications where the problem justified it
- Established reusable boilerplates, conventions and CLI tooling that made project setup faster and maintenance predictable over time
- Led technical scoping, estimation and client-facing technical advisory through discovery and delivery
- Took part in technical hiring and onboarding, and set the standards new developers worked to

Representative clients: Unicef · Oxfam · Engie · Delhaize · Telenet · KBC Brussels · Latour & Petit · Cefic · Business Europe · Clear Channel · Monizze · Solidaris · eFarmz · RTL · RTBF

SELECTED ENGINEERING WORK

stephan.zych.be, two runtimes from one content source. One Markdown/JSON content source feeds two independent runtimes: an Eleventy/Lit/TypeScript web application and a Go SSH/TUI application. GitHub Actions validates the repository, builds both container images separately, publishes them to GHCR and deploys automatically to DigitalOcean.
github.com/monkeymonk/stephan.zych.be

Developer tooling around coding agents. A Go tool renders one canonical source (agents, skills, MCP servers, rules and reusable instructions) into the configuration of several coding-agent CLIs, so the same engineering rules apply whichever tool is driving; prompt.nvim carries those workflows into the editor. Agent

work runs on deterministic checks, reviewed diffs and Git worktrees, with sandboxing (Bubblewrap, Landlock, seccomp) kept outside the agent's control.
stephan.zych.be/blog/ai-agent-sandboxing/

EARLIER ROLES

Freelance Web Developer — Independent · 2011 – 2016

Rich Media Developer — TBWA Group · 2012 – 2013

Frontend Developer — Agency.com · 2011 – 2013

Co-Founder / Rich Media Developer — La Cerise Numérique · 2010 – 2013

Web Designer / Integrator — Com2B Networks · 2009 – 2011
Representative clients: IKEA · L'Oréal · Nespresso · Sony Pictures · bpost · European Parliament

TECHNICAL STACK

Application & frontend

PHP · Laravel · WordPress · Bedrock / Sage · JavaScript · TypeScript · Node.js · Go · React · Next.js · Vue · Nuxt · Angular · Lit & Web Components · Tailwind CSS

Platform & delivery

Docker · Docker Compose · Linux (Ubuntu) · Traefik · Caddy · Nginx · GitHub Actions · GitLab CI · GHCR · Automated deployments

Data & infrastructure

MariaDB / MySQL (primary) · Redis · PostgreSQL · SQLite · AWS Lightsail · DigitalOcean · Cloudflare

Engineering practices & observability

Git · Conventional Commits & SemVer · Grafana · Prometheus · Loki · GoAccess

AI-assisted engineering

Claude Code · Codex CLI · OMP · MCP / agent configuration · Ollama · llama.cpp · Agent sandboxing (Bubblewrap, Landlock, seccomp)

EDUCATION & TRAINING

Académie Royale des Beaux-Arts de Bruxelles — Bachelor: Plastic, Visual & Spatial Arts

Université Libre de Bruxelles — Computer Science studies

École de Recherche Graphique (ERG) — Visual Narration studies

CEPEGRA — Graphic production training

LANGUAGE SKILLS

French — C2 · Native · English — B2

LEADERSHIP & COMMUNITY

Internal tech talks · CBTW — “Terminal utilities & advanced use cases”, “Conventions & patterns in production”

Member, Réseau Entreprendre Bruxelles — 2019 – 2023