

ANDREI AMZA

Full-Stack Software Engineer | TypeScript · NestJS · React · AWS

✉ andrei.amza177@gmail.com 📞 +41 79 921 72 38 📍 Zürich, Switzerland

🌐 andreiamza.dev 📄 github.com/amzaandrei

PROJECTS

AutoApply

AI job-application platform.

A job-search tool I'm building: it finds roles, drafts a tailored cover letter for each, and tracks the applications and interviews. **Anthropic SDK** does the writing, **tRPC** and **BullMQ** + **Redis** run the background work, **Stripe** and **NextAuth** on top.

[Next.js 16](#) · [React 19](#) · [TS](#) · [Prisma](#) · [PostgreSQL](#)

SlidePrep

Turn slides into personalised quizzes.

Upload a PDF or slide deck and it pulls out the key ideas, then writes exam-style questions on them. I built the ingestion over **pdf-parse**, **pdfjs-dist** and **officeparser** and let **Claude** do the extraction and question-writing. Multi-tenant with **Stripe** tiers.

[Next.js 16](#) · [React 19](#) · [TS](#) · [Prisma](#)

SEC 10-K Analyzer

RAG Q&A over SEC filings.

A CLI that pulls 10-K filings from EDGAR, embeds them locally and answers questions about them with citations back to the source. Embeddings run on-device (**Transformers.js**), **FAISS** for search, **Claude** for the answers, plus per-query cost tracking.

[Node.js](#) · [TS](#) · [Anthropic SDK](#) · [Transformers.js](#) · [FAISS](#)

[LangChain](#) · [Cheerio](#) · [Commander](#)

SKILLS

I work mostly in **TypeScript** across the stack — **Next.js**, **React 19**, **NestJS** and **Node.js** — with type-safe APIs over **tRPC**, **GraphQL Federation** and **Zod**. A lot of my recent work is LLM-based, using the **Anthropic** and **OpenAI** SDKs and **MCP** servers; for RAG pipelines I reach for **LangChain**, **FAISS** and **Transformers.js**. I've shipped iOS and cross-platform mobile in **Swift**, **SwiftUI**, **UIKit** and **Kotlin Multiplatform**. For data I use **Prisma**, **PostgreSQL**, **MySQL** and **Redis/BullMQ**. I deploy to **AWS** (**ECS**, **Lambda**, **SQS**, **CloudFormation**) and **GCP**, or self-host with **Docker** + **Caddy**, and manage monorepos with **Turborepo** and **pnpm**. I set up CI/CD in **GitHub Actions** and **Fastlane**, instrument production with **Sentry** and **PostHog**, and build UIs with **Tailwind**, **shadcn/ui**, **Radix**, **Material-UI** and **Vite**. On the CLI side I'm comfortable with **Cheerio**, **Commander** and **Python**; version control is always **Git**.

LANGUAGES

Romanian	Native
German	C1
English	C1

EXPERIENCE

Full-Stack Developer — Social Unlimited GmbH

Jan 2025 – Mar 2026 · Thaur, AT

- Helped build **eCom360**, a platform that ties together our backend services, external systems and admin tools in one domain-driven monorepo, so the team works against one consistent codebase instead of a handful of disconnected projects.
- Put a single **GraphQL Federation** gateway in front of the **NestJS** services, giving the frontend one API to call instead of stitching several together; services coordinate asynchronously over **AWS SQS** + **CloudEvents**, so a slow external system never blocks the rest of the flow.
- Wrote a reusable adapter layer for external integrations — including a two-way **Shopify** product sync — which turned “add the next integration” into implementing one interface rather than rebuilding the plumbing each time.
- Built the admin portal as a single-page app in **React Router 7** on a shared **Material-UI** component library with **AWS Amplify** auth, giving a consistent UX across the internal apps.
- Set up the whole delivery path: containerised deploys to **AWS ECS Fargate** via **CloudFormation** and CI/CD in **GitHub Actions**, so changes ship automatically once tests pass.

iOS Developer — M-Pulso GmbH

Mar 2023 – Dec 2024 · Innsbruck, AT

- Built the **white-label iOS** app that several clients shipped under their own branding — **Swift** + **CMFCore** with a theming and asset system, so launching a new client app became a configuration change rather than a fresh codebase.
- Shared the business logic between iOS and Android with **Kotlin Multiplatform**, cutting roughly 40% of the duplicated code and keeping the two apps in sync as features landed.
- Automated signing, certificates and releases with **Fastlane**, which brought a single release down from days of manual work to a couple of hours.
- Wrote the real-time chat over **WebSocket** (1:1 and group rooms) with offline-first storage, message pagination and unread badges, holding up under thousands of concurrent users.

Cloud Developer — Universität Klagenfurt

Apr 2022 – Oct 2022 · Remote

- Worked on **CardioHPC**, a cardiac-monitoring pipeline that flags heart-attack risk; my part was reworking the server-side processing into a **Function-as-a-Service** architecture, so it scaled with incoming load instead of sitting on fixed, over-provisioned servers.
- That work grew into my BSc thesis: a multi-cloud scheduler that spreads workflows across **AWS** and **GCP**, weighing runtime against cost — it would pick the cheapest way to hit a deadline, or the fastest way to stay inside a budget.

iOS Developer — S.C. Twelve IT

Sep 2019 – Aug 2021 · Romania

- Built and maintained production iOS apps in **Swift**, **SwiftUI** and **UIKit**, wired up to **AWS** and **GCP** backends.
- Structured features around MVC/MVVM and wrote reusable **Swift Package** API clients, which cut repeated networking code across the apps and made new screens quicker to build.
- Kept regressions out with **XCTest/XCUITest** coverage and tracked down the harder production bugs through systematic debugging.

EDUCATION

BSc Computer Science — University of Innsbruck

2019 – 2023 · Innsbruck, AT

Thesis: **MatchFaaS** — a matching-based scheduler that splits serverless workflows across **AWS** and **Google Cloud** regions to get past per-region concurrency limits, cutting workflow runtime by roughly 2–2.5× on the evaluated workloads.

High School Diploma, ICT — Grigore Moisil

2015 – 2019 · Braşov, RO