

Kevin Deveau

Full-Stack & Systems Engineer — Compilers, Production SaaS, ML Infrastructure

+1 (403) 363 9396 | softwaredev.eau@gmail.com | devkev.dev | [LinkedIn](#) | [GitHub](#)

EDUCATION

University of Alberta

Bachelor of Science in Computer Science with Specialization

Relevant Coursework | Compiler Design & Optimization, CPU & GPU Architecture, Machine Learning & AI, Data Structures & Algorithms, Software Engineering, Operating Systems, Databases, Project Management, Probability & Statistics, Linear Algebra.

Edmonton, AB

Expected December 2026

EXPERIENCE

Founding Software Engineer

2025 – Present

Lustr Services (Digitally-Powered Cleaning Startup)

Edmonton, AB

- Shipped a production full-stack platform on **React (Vite)**, **Supabase/Postgres**, **Stripe**, and **Vercel** running real customer bookings, recurring memberships, and admin CRM.
- Engineered a conflict-aware **scheduling engine** with multi-crew capacity, interval overlap, and **10-minute slot holds** enforced by **SELECT FOR UPDATE** row locks; eliminated double-booking under concurrent checkout load.
- Architected the production **Postgres** backbone: **30 tables**, **95 RLS policies**, **55 stored procedures**, **30 edge functions**, with forward-only migrations gating Vercel promotion against half-migrated schemas.
- Built a **CI/CD release pipeline** with three-tier branching, two PR gates, hotfix auto-backport, and **100+ Playwright E2E** + **250+ Vitest** tests across booking, payments, scheduling, and auth.

IT Support

2022 – Present

Givergy

Edmonton, AB

- Deployed and operated live event-tech stacks (**tablet bidding fleets**, networking, auction software, payment terminals) at fundraising events; diagnosed and resolved live software/hardware faults under time pressure to keep **100% uptime** on bidding and payment processing.
- Partnered with event staff to triage incidents in real time, cutting issue resolution time by **25%** across high-traffic auction nights.

PROJECTS

Azimuth – Flight Club & FBO Management Platform

Next.js 16, React 19, PostgreSQL, Drizzle ORM, Auth.js v5

- Built a **multi-tenant SaaS** replacing dated FBO software: six operational modules (members, fleet & maintenance, scheduling, dispatch, flight logging, billing) over a **~60-table Drizzle/Postgres** schema (44 migrations), org-scoped from day one so a second club needs no schema change.
- Implemented **capability-based access control** (6 roles → fine-grained capabilities) checked on every server action; **Next.js 16** Server Actions, **Auth.js v5** JWT sessions, reservation conflict detection, and Vercel Cron reminder automation.

Gazprea Language Compiler

ANTLR4, MLIR, LLVM, C++17 | 4-person team

- Built a **9-pass compiler frontend** (scope, type resolution, qualifier enforcement, type checking, cast validation) over a 342-line **ANTLR4** grammar; polymorphic **TypeInfo** hierarchy with chained-alias cycle detection across **40+ language constructs**.
- Designed **MLIR** codegen across **6 dialects** (Arith, SCF, Func, MemRef, Math, ControlFlow) lowered through a 7-stage pipeline to **LLVM IR**; split codegen into 9 modular services (LoopEmitter, MatrixService, SubroutineEmitter) composed via **BuilderScope RAII**.
- Authored **2,101 integration tests** in GitHub Actions on a self-hosted container, diffing **stdout** byte-for-byte; finished **#1 in course** and presented in an academic exchange with **HSBI Germany**.

Tile-Level ML DSL Compiler

ANTLR4, MLIR, LLVM, NVPTX, C++17

- Designed a **Triton/Halide-style tensor DSL** with first-class tile types (**Tile<f32, [BM, BK]>**) and a **5-pass static type checker** emitting clang-style diagnostics for shape, layout, and bind-dimension errors at compile time.
- Authored a custom **tile MLIR dialect** (ODS/TableGen) with **dual backends**: a CPU path lowering through **linalg/memref/scf** to **x86-64 AVX2** assembly, and an **NVIDIA sm_80 GPU** path with memory-space-aware lowering (**cp.async**, **ldmatrix**, **nvgpu.mma.sync** tensor cores).

BookNook – Multilingual Reading Platform for SFU Linguistics

FastAPI, Next.js 14, PostgreSQL, Meta NLLB, Docker | 7-person team

- Served as **product owner** on a 7-person agile team delivering the research output for **SFU Linguistics**; shipped **300+ user stories** across 5 sprints, supporting K-12 reading in **200+ languages** including low-resource ones (Pashto, Somali, Amharic).
- Architected a **FastAPI** backend (14k+ lines, SQLAlchemy) with a **5-stage async PDF pipeline** (OCR, review gate, translation); integrated **Meta NLLB** with singleton lazy-loading to cut cold-start from **20s to near-zero**; shipped a **Next.js** front end (WCAG, PWA) with **88 test files at 80% coverage** and GitHub Actions CI/CD.

TECHNICAL SKILLS

Languages - C/C++, Python, Java, TypeScript/JavaScript, SQL, Dart, R, HTML/CSS. Assembly: RISC-V, ARM, x86

Compilers & Systems - MLIR, LLVM, ANTLR4, NVPTX (tensor cores, **cp.async**, **ldmatrix**), AVX2, CMake, lit/FileCheck, GDB

Frameworks & Tools - React, Vite, Next.js, FastAPI, Node.js, Tailwind, Drizzle ORM, Docker, Git, GitHub Actions, Vercel, Playwright, Vitest, Pytest

Data & Platforms - PostgreSQL, Supabase (RLS, RPCs, Edge Functions), MySQL, SQLite, MongoDB, Firebase, Stripe, PyTorch, NumPy/Pandas