

AHMED M. SOZZER

ahmed@amsozzer.com | amsozzer.com | github.com/Amsozzer1 | linkedin.com/in/amsozzer1 | Austin, TX

Education

University of Illinois Urbana-Champaign

Bachelor of Science in Computer Science

May 2025

Experience

FYCLabs

Remote

Full Stack Engineer (Engineer II)

May 2025 – Present

- Work directly with client founders, C-suite, and end users to scope requirements and surface pain points, then translate them into shipped features
- Build and ship full-stack applications for early-stage startups end-to-end, taking product concepts to production with Next.js, React, TypeScript, PostgreSQL with Prisma, and GCP infrastructure
- Build cross-platform mobile apps in React Native from Figma designs, owning the full API layer and database modeling
- Engineer CRM integrations against client backends (GoHighLevel, Zoho, HubSpot) via REST APIs and webhooks, replacing manual data entry for client operations teams
- Build automated CI/CD pipelines on GCP with test gates, taking releases from manual to push-button across multiple concurrent client projects
- Optimize backend and database performance to scale from 700 to 5000 active users, improving query efficiency and horizontal scalability across distributed GCP infrastructure

Holiday Channel

Colorado Springs, CO

Full Stack & iOS Developer

Jan 2025 – May 2025

- Built and shipped an e-commerce platform end to end, from checkout flow through order fulfillment
- Shipped the customer-facing mobile app in React Native for iOS, from Figma designs through App Store release
- Designed the service architecture to absorb holiday-season traffic peaks without degrading checkout
- Integrated Google Ads and built responsive interfaces from Figma designs using Shadcn and Framer Motion

Luminii LLC

Niles, IL

Data Analyst and Software Intern

May 2024 – Aug 2024

- Built an ML-driven pricing system on real-time supplier data to adjust component costs, replacing a manual re-calibration process
- Reindexed a 15,000+ record database and added automated data validation, cutting query times across the pricing pipeline

Skills

Languages: Python, C/C++ (C++17), JavaScript/TypeScript, Java

Frameworks: React, Next.js, Node.js, Express, React Native, FastAPI, TailwindCSS

Systems: POSIX sockets, libuv, MQTT, multithreading, CMake, embedded (ESP32), Linux

Databases & Cloud: PostgreSQL, Prisma, MongoDB, Redis, GCP, Firebase, Docker

Practices: REST APIs, Webhooks & Integrations, CI/CD, System Design, Testing (GoogleTest, Jest, Cypress)

Projects

AMS-X — Open Modular Filament System for Bambu Lab Printers | [Repo](#)

- Bambu Lab printers cap multi-material printing at 4 spools, forcing a manual swap on every material change; AMS-X removes the cap by running the printer in external-spool mode and automating the swap over MQTT (Python, FastAPI, ESP32) — protocol is undocumented, so the command set and report fields were reverse-engineered against a live printer
- Server owns the job: parses the sliced 3MF into a material-swap plan, pushes it over LAN FTPS, starts the print over MQTT, and validates every live pause event against the plan before anything moves
- Swap state machine (retract → select → feed → sense → resume) drives the printer's own change routine rather than authoring hotend gcode; the spool module is an interface with a human implementation today and a stepper (TMC2209 over ESP32) next

PlusWeb — Express-style HTTP Framework in C++ | [Repo](#)

- Built an Express-style C++17 HTTP framework (app.GET, app.use) on a libuv event loop and the llhttp parser; 4.8x Express on a small app and flat from 5 to 10,000 routes where Express degrades 50x. MIT-licensed and installable via CMake
- Designed a segment-trie router keyed by METHOD:/path with parameterized routes (/users/new beats /users/:id), plus mountable sub-routers
- Implemented a composable middleware chain with next() continuations, path-scoped middleware, and relative-path rewriting for nested routers
- Unit and integration tested (GoogleTest/CTest), the integration suite booting a live server and driving it over loopback; CI covers Linux and macOS, verifies the install target, and re-runs the suite under AddressSanitizer and UBSan with leak detection

Publications & Presentations

- Contributed to the mentor-mentee matching research; credited in the acknowledgements of the ASEE work-in-progress paper ([Link](#))
- Presented the mentor-matching research at the 2023 John S. Albazi Student Research Symposium ([Link](#))
- Presented the mentor-matching app research at Wright College's Fall 2022 Student Research Colloquium ([Link](#))