Arnav Jindal

(919) 914-3630 contact@arnavjindal.xyz arnavjindal.xyz github.com/daggy1234

Education

Duke University **Aug. 2022 – May 2026**

Bachelor of Science in Computer Science (Software Systems) and Economics (Finance) *Durham, NC*

- **GPA:** 3.6
- **Relevant Coursework:** Data Structures, Databases, Computer Architecture, Discrete Mathematics, Microeconomics, Macroeconomics, Probability, Multivariable Calculus, Linear Algebra, Computer Graphics, Analysis of Algorithms, Audio & Image ML, Functional Programming, Asset Pricing & Risk, Econometrics, Operating Systems
- **Organizations:** Computer Architecture Teaching Assistant, Duke Quant Finance, HackDuke, Duke Applied ML

Experience

IMC Trading **June 2025 – August 2025**

Software Engineering Intern – D1 Execution *Chicago, IL*

- Built a distributed Java Apache Flink cluster to statefully process over 100 million messages per day via Kafka Streams, deployed on Kubernetes and replaced a failing service with 60% lower latency, 2x throughput and 100% accuracy.
- Analyzed C++ engine code and feed data packets to improve handling for MASS_CANCEL orders, order reductions and order lifecycle tracking on trading engines. Worked with traders to analyze and fix certain strategies.

Alleviate Health **Aug. 2024 – May 2025**

Founding Engineer *Remote*

- Built full-stack dashboards and agentic texting/voice workflows that screened patients automatically and booked appointments. Scaled systems to handle over 200K texts and 100K patient interactions.

Amazon, Inc. **May 2024 – August 2024**

Software Engineering Intern *Seattle, WA*

- Developed an internal command-line tool, distributed via Amazon's package manager, that streamlined accessibility testing by triggering automated UI scans, reducing manual testing time by 40%.
- Engineered AWS infrastructure to optimize scanner Lambdas through WebSocket proxies, resulting in a 50% reduction in scan times and a 25% decrease in operational costs.
- Implemented blue-green deployment strategies and CI/CD pipelines with integrated alarms and metrics collection, enhancing observability and reliability of the production stack.

TrendUp AI (YC S23) **June 2023 – August 2023**

Software Engineering Intern – Full Stack *Remote*

- Programmed vector stores to scrape and ingest datasets for ChatGPT querying via APIs. Developed user flows to create and post content powered by TrendUp's AI tools.
- Deployed backend pipelines to ingest and process data in the background, reducing compute costs by 30% and expanding data storage by 4x.

Duke University Dept. of Computer Science **June 2023 – August 2023**

Research Assistant – Computer Systems *Durham, NC*

- Developed mRPC with Dr. Matthew Lentz and Dr. Danyang Zhuo, a managed and efficient gRPC alternative featuring centralized Rust proxies and improved marshaling for 100% faster RPC calls.
- Engineered a user-friendly Python frontend using Pyo3 and Rust FFI; authored documentation with examples and co-authored a research paper detailing findings.

Projects

Contract Exchange – Click Trading Simulator: Built a distributed backend exchange with an order book, position tracking, and support for leveraged trade types. Enforced price–time priority with RabbitMQ and ensured reliability with SQLite and Redis. Developed a trading UI with WebSockets and visualizations for Duke's trading competition; used by over 100 teams.

Dagpi – Full Stack SaaS Platform: Web app featuring Next.js frontend and TypeScript/Express.js backend. Image manipulation microservice in Python/FastAPI, authentication and metrics microservices in Actix (Rust). Deployed on Docker/Kubernetes with Nginx, monitored by Grafana and Prometheus. PostgreSQL and TimescaleDB with AWS Glacier backups; Stripe and PayPal payment gateways.

Polaroid – Python Image Library: Integrated Rust libraries via Pyo3 FFI to deliver faster filesystem access, I/O operations, and pixel matrix computations. CI pipeline builds cross-platform wheels using Maturin and auto-generates documentation.

Esoteric Languages – Niche DSLs: Built two languages. Udymts: a Rust-based CLI/compiler that emits Python. PokeType: an expression evaluator with rply-based lexer, compiler, and interpreter.

Graphics Engine – C++ CPU Rendering: Developed a 2D graphics engine for pixel-level rendering, geometric primitives, blending, clipping, Bezier curves, and matrix-based transformations. Implemented shader tiling and fragment shaders. Tuned performance using profiling, template meta-programming, loop unrolling, and function lookup tables.

Technical Skills

Languages: Java, C++, Python, Rust, TypeScript, JavaScript, SQL, Bash, Deno, MIPS, Solidity
Libraries/Frameworks: Node.js, React, Flask, Actix, perf, valigrind, postrgeSQL, bazel, Jupyter, langchain
Tools/Platforms: Git,AWS, Docker, Kubernetes, Cloudflare, Vercel, Grafana, Prometheus, RPC, MCP, Vector Stores