

Avinav Bhandari

avinavbhandari@gmail.com | 587-575-3424 | Toronto, ON
avinav.dev | linkedin.com/in/avinav-bhandari | github.com/avinav-b

Education

University of Toronto

B.A.Sc. Computer Engineering

GPA: 3.7/4.0 · Minor in Artificial Intelligence Engineering

Coursework: Data Structures and Algorithms, Operating Systems, Computer Systems Programming, Deep Learning, Computer Organization, Digital Systems, Computer Graphics, Probability

Toronto, Canada

Expected May 2027

Experience

Software Engineering Intern - AI & Full Stack

BiblioCommons

May 2025 – Present

Toronto, ON (Remote)

- Architected a RAG platform (Spring AI, Java, pgvector) over **50,000+** documents, reaching **70%** company-wide adoption and cutting reported retrieval time **80%**.
- Cut p95 latency up to **50%** on a REST API serving **20M+** users by optimizing database queries and business logic.
- Built an agentic AI SRE assistant with tools over Sumo Logic, Prometheus, Checkmk and Confluence/Jira APIs plus a self-updating incident history; cut resolution time **50%+** and doubled the on-call pool.
- Triaged live outages and shipped hotfixes even as a co-op/intern.
- Built WCAG 2.2-compliant React features for libraries and patrons that passed third-party accessibility audits.
- Cut page load times up to **30%** in legacy Rails and Java services by rewriting business logic for efficient queries.
- Closed **100+** production bugs for **250+** library customers (logic, ActiveMQ retries, ILS API sync), extending ILS integrations where needed; praised by support for speed and clarity.
- Raised RAG retrieval accuracy to **95%** on a labelled question-to-source eval set by adding retrieval agents that re-query when context is missing.
- Exposed the RAG platform as an internal API for other engineers' tooling, adopted by another team to query Zendesk docs.

Co-Founder and Lead Engineer

Rheely

Jan 2026 – Present

Toronto, ON (Remote)

- Co-founded a media provenance platform and owned all engineering, taking it from pre-revenue to **2+** enterprise news customers.
- Engineered a capture-integrity pipeline (custom ML models, Apple App Attest, Play Integrity, device signals) detecting emulators, replays and recaptured media.
- Shipped a React Native/Expo app (iOS/Android) with native Kotlin/Swift modules for capture, pixel signing and motion sensing; 0 to **1,000+** users.
- Stood up the Django backend and infrastructure from scratch (Postgres, Redis, S3, GitHub Actions CI/CD, Sentry), serving **1,000+** users.
- Pitched Rheely's product and technology to prospective investors and enterprise customers.
- Built the product end to end with AI coding agents (Claude Code, Cursor, T3 Code) as a core part of the engineering workflow, roughly tripling engineering throughput.
- Wrote the technical specification that passed external C2PA conformance assessment, placing Rheely on the official C2PA conformance list.

Software Engineering Intern - Full Stack

Draft Party

May 2024 – Aug 2024

Remote

- Built a daily game mode on Cloudflare Workers and Durable Objects for a trivia platform with ~100 players and corporate clients.
- Built **20+** TypeScript/Node.js REST endpoints for accounts (signup, email verification, password reset, JWT), tested with Jest.
- Built the Vue/Nuxt account dashboard and JWT login flow, including password resets and daily-game email subscriptions.

Jr. Part-time Frontend Software Developer

Draft Party

Jan 2024 – May 2024

Remote

- Rebuilt the website from UI/UX mock-ups in Vue/Nuxt and TypeScript, with responsive layouts for **800+** players.
- Built a new leaderboard page that helped win several new corporate customers shortly after release.

Event Software and Livestream Lead

Mecha Mayhem

Feb 2022 – Present

Calgary, Canada

- Lead a 6-person team building a React Native/Expo app used by **400+** competitors each year.
- Build React event-management software used by **100+** staff to run the competition and track volunteers.
- Built real-time livestream graphics driven by live match data over MQTT and WebSockets, seen by **5,000+** viewers.

- Serve on the central leadership team running Canada's largest robotics tournament, entirely volunteer-run.
- Released tm-switcher (TypeScript/Electron), automating OBS livestreams from VEX Tournament Manager; adopted by other VEX events.

Avionics/Software Developer

Sep 2022 – May 2024

University of Toronto Aerospace Team - Rocketry Division

Toronto, Canada

- Developed avionics software on a ~20-person hybrid-rocket team, placing 3rd (2023) and 2nd (2024) at Launch Canada among **30+** teams.
- Wrote a headless C/C++ Linux app on a Raspberry Pi 4 driving GPIO to run the fuelling system; fuelled the rocket successfully on launch day.
- Led development of a C++ backend that parsed thermocouple data and transferred it reliably over a small local network.
- Built Python tools to parse and store GPS/GNSS telemetry, enabling live tracking of the rocket's position from the ground.

Projects

Crumbz Recipe App: Recipe Searcher | *Personal project*

May 2024 – Present

- Built Crumbz, a React Native (iOS/Android) recipe app that matches the ingredients users have on hand to **1,000+** recipes they can cook.
- Built a Go REST API with **25+** endpoints and JWT authentication on PostgreSQL, containerized with Docker and deployed on Google Cloud Run.
- Built ingredient matching that combines string-matching algorithms with an agentic LLM pipeline for semantic reasoning over recipes.

Complete Breadboard CPU and Custom Instruction Set | *Personal project*

May 2024 – May 2025

- Designed and built a working 8-bit CPU on breadboards from logic-gate ICs, with a variable-speed clock.
- Built an ALU supporting addition, subtraction, shifts and logic operations.
- Implemented memory, a register file and a program counter, and designed a custom 13-instruction ISA with bespoke opcodes to load and run multi-step programs.

Rocket Projectile Simulator | *ECE243 course project, University of Toronto*

Jan 2024 – May 2024

- Built a physics-based projectile simulator as a rocket game in C/C++, running bare-metal (no OS) on a Nios II soft processor on a DE1-SoC FPGA board.
- Wrote C drivers for a 640x480 60 Hz VGA display with double buffering.
- Wrote a PS/2 keyboard driver for player input.

FPGA Human Benchmark Games Project | *ECE241 course project, University of Toronto*

Sep 2023 – Dec 2023

- Built Human Benchmark-style reaction-time and chimp-memory games entirely in Verilog/SystemVerilog on a DE1-SoC FPGA.
- Implemented a double-buffered VGA display pipeline and PS/2 mouse input with an on-screen cursor for point-and-click gameplay.

Skills

Languages: Python, Java, TypeScript/JavaScript, Go, C/C++, C#/.NET, Kotlin, Swift, SQL, Verilog/SystemVerilog

Frameworks: React, React Native/Expo, Next.js, Vue/Nuxt, Node.js, Django, Spring AI, Ruby on Rails, Electron, Tailwind CSS, Jest

AI/LLM: RAG, Agentic LLM pipelines, pgvector, Re-ranking, Evals

AI tools: Claude Code, Cursor, T3 Code

Data & Infrastructure: PostgreSQL, Redis, ActiveMQ, AWS S3, Google Cloud Run, Cloudflare Workers/Durable Objects, Docker, GitHub Actions, Sentry, Linux

Other: REST APIs, JWT auth, MQTT, WebSockets, Git/GitHub, Jira, Raspberry Pi/GPIO