

Eva Liu

evaxliu02@gmail.com | github.com/evaxliu | [linkedin.com/in/el02/](https://www.linkedin.com/in/el02/)

EDUCATION

Georgia Institute of Technology, School of Computational Science & Engineering Incoming Spring 2027

- Masters of Science in Computer Science

University of Washington, Paul G. Allen School of Computer Science & Engineering Sep 2020 – Dec 2024

- Bachelor of Science in Computer Science

TECHNICAL SKILLS

- TypeScript (NextJs, React), Python, HTML, CSS, Supabase, Fast API, Voyage AI embeddings, pgvector, and semantic search

SOFTWARE EXPERIENCE

Z&V Home Construction LLC, Frontend Software Engineer Contract Jun 2026 – Present

- Rebuilt the company's Wix site as a custom responsive **Next.js** application in **React, TypeScript, and Tailwind CSS**, roughly tripling daily traffic within two weeks of launch and reaching an 11.6% search click-through rate, with a project gallery, service sections, and a lead capture contact form.
- Owned deployment and discoverability end to end, configuring hosting, a custom domain, HTTPS, and automatic builds from source control, then implementing page metadata, Open Graph tags, and a sitemap and creating the business's first Google Search Console property to get the site indexed and measurable for local search.
- Maintained the live site as the sole developer, handling content updates, image and performance optimization, dependency upgrades, and new feature requests.

Center for Reproducible Biomedical Modeling (CRBM), Software Engineer Jul 2022 – Dec 2024

- Led a team of student engineers building a browser-based computational biology modeling editor, letting researchers create models with zero local setup.
- Owned end-to-end execution across roadmap, sprint planning, and delivery, converting requirements into shipped features.
- Presented at international standards conferences (COMBINE, HARMONY) and co-authored a peer-reviewed Oxford Bioinformatics paper on the VSCode Antimony extension (DOI: 10.1093/bioinformatics/btad753).
- Built a React/TypeScript in-browser editor for the Antimony modeling language on Monaco Editor, implementing grammar recognition, syntax highlighting, and contextual hover documentation.
- Developed core features for the VSCode Antimony extension (6,000+ Marketplace downloads), including variable annotations and dynamic rate law insertion via VS Code APIs, cutting manual authoring effort for modelers.
- Engineered an automated Python and TypeScript installation pipeline, removing setup friction for new users.

SIDE PROJECTS

[Blog Platform](#) Oct 2025 – Present

- Built a full-stack blogging platform using **Next.js, React, TypeScript, Python, HTML, CSS and Supabase with Wisp CMS** for structured content management and dynamic rendering.
- Developed an AI-powered article recommendation system using **Voyage AI embeddings, pgvector, and semantic search** to surface related articles.
- Developed backend services with **FastAPI** to generate, store, and retrieve vector embeddings to improve user content discovery.
- Implemented modular frontend architecture to support reusable layouts and scalable content delivery.

[Find Your Needs](#) Mar 2024 – Jun 2024

- Built a cross-platform Flutter/Dart app (iOS and Android) surfacing food, hygiene, and library resources across greater Seattle, rendering a typed location dataset as icon-differentiated Google Maps markers alongside a distance-sorted list view with deep-link handoff to the device's native navigation app.
- Implemented real-time geolocation via a Provider state layer feeding live distance and marker updates, with session persistence for map position and zoom through shared preferences.

[Pet My Dog](#) Mar 2022 – Jun 2022

- Built a multi-page React SPA with client-side routing that lets dog owners broadcast real-time location and find nearby users for in-person meetups.
- Implemented continuous geolocation tracking via the browser Geolocation API, throttling position updates to the server and syncing map and list views through React state and effect hooks.
- Wired controlled form inputs through async fetch calls to REST endpoints (GET for nearby users, POST for location and profile writes), with input validation, HTTP error handling, and database persistence.