

Julius Liu

Full Stack Developer | TypeScript, React, Next.js

Email: CONTACT@JULIUSLIU.DEV Portfolio: PORTFOLIO.JULIUSLIU.DEV

EDUCATION

Computer Programming Diploma, Algonquin College, Ottawa ON

Aug 2026 (Expected)

- Cumulative GPA: 3.98/4.0, Dean's Honours List
- Relevant coursework: object-oriented programming in Java, system design and analysis, database management and administration, web development with PHP, computer networking

SKILLS

- **Backend:** JavaScript, TypeScript, React, Node.js, Express, Hono, Next.js, Zod, Postman, Vitest, Playwright, Flask, Django, PHP
- **Frontend:** HTML, CSS, Bootstrap, Tailwind CSS, shadcn/ui, Framer Motion
- **Database:** MySQL, PostgreSQL, MongoDB, Prisma, Drizzle, GraphQL, SQL Server, Oracle
- **Other languages & tools:** Java, JUnit, Go, Python, Git, Husky, GitHub Actions, Docker
- **Domain knowledge:** data structures and algorithms, operating systems, AWS, automated testing

PROJECTS

Morph (AI-powered essay editing)

Next.js, MongoDB, Prisma

- Integrated **Gemini API** with OpenAPI schema and prompt engineering for **structured output**.
- Developed **accessible, responsive design and consistent UI** with Tailwind CSS and shadcn/ui.
- Incorporated **cache and data preloading** for better user experience, **lowering FCP by 80%**.
- Ensured deployment reliability with a **robust CI/CD pipeline** featuring **end-to-end testing** using Github Actions and Playwright.

Shuffle (note-taking tool with text highlighting)

Next.js, Hono, SQLite, Drizzle

- Designed a **normalized relational database**, implemented **user authentication**, securing **RESTful API endpoints** with **middleware-based optimistic authentication and authorization** checks.
- Delivered client-side data management using SWR for fetching, caching, and optimistic updates.
- Achieved **end-to-end type-safety** with **client/server validation** (Zod and React Hook Form) and Hono RPC client integration.
- Created a **custom React hook** and components for reliable and efficient retrieval, processing, and rendering of text selections.

Gitlet (object-oriented Git implementation)

Java

- Built a simplified Git-like system supporting **essential version control commands** (init, add, commit, checkout, merge, etc.) following modular and object-oriented design in Java.
- Persisted repository state (branch heads, staging areas, commit references) via **serialization**.
- Implemented an **efficient merge-base algorithm** for branch merging with **conflict resolution and comprehensive error handling**.