

Weifan Li

540-998-5718 | li.weifa@northeastern.edu | github.com/lhq5520 | Vancouver, BC (Open to relocate for co-op)

EDUCATION

Northeastern University <i>Master of Science in Computer Science; GPA: 3.9/4.0</i>	Vancouver, BC <i>Jan. 2024 – Dec. 2026</i>
Virginia Polytechnic Institute and State University <i>Bachelor of Science in Human Nutrition, Foods, and Exercise; GPA: 3.7/4.0</i>	Blacksburg, VA <i>Aug. 2018 – Jan. 2022</i>

EXPERIENCE

Backend Web Developer Intern <i>Qihoo 360</i>	Oct. 2022 – Jan. 2023 <i>Beijing, China</i>
<ul style="list-style-type: none">Built Spring Boot microservices (user management, data-source registry, access control) with secure REST APIs and role-based authentication, improving API reliability during peak internal traffic.Optimized MyBatis + MySQL query performance and schema usage, reducing slow-query latency by ~25% for high-volume data operations.Refactored fault-tolerant logic and improved logging for an internal job scheduler, cutting recurring workflow failures by ~40%.Exposed endpoints for automated asset cataloging and reporting across Hive/TiDB/MySQL, reducing manual data-processing workload for analytics teams by ~30%.	

PROJECTS

Mountify — E-commerce Platform <i>Next.js, TypeScript, PostgreSQL, Stripe</i>	2025
<ul style="list-style-type: none">Built a full-stack e-commerce platform using Next.js, TypeScript, PostgreSQL, and Node.js, supporting product catalog, cart/checkout flow, secure sessions, and an admin dashboard.Upgraded the order status system from REST polling to an event-driven webhook pipeline, reducing unnecessary API traffic and enabling reliable 1–2s payment confirmations.Hardened backend security by enforcing server-side price, inventory, and permission validation, replacing early client-trusted logic with strict input checks and DB-backed verification.Integrated Stripe Checkout with server-side payment intents, webhook-driven order verification, and idempotent charge handling to prevent duplicate payments.Designed and optimized a normalized PostgreSQL schema using indexes, foreign keys, and transactional updates; improved read-heavy endpoints to less than 150ms.	

Personal Blog <i>React, Node.js, MongoDB, flask(legacy)</i>	2024
<ul style="list-style-type: none">Originally built the blog as a Python Flask app with templated pages and SQLModel, then migrated it to a modern React + Node.js/Express SPA while preserving core features and improving separation of concerns.Developed a full-stack blog using React and Node.js/Express with secure JWT authentication, session handling, and end-to-end CRUD workflows.Designed and implemented REST APIs with access control and Mongoose-based data validation, cutting invalid request errors by ~40%.Built a responsive UI using React Router and custom hooks; integrated a rich text editor and optimized MongoDB queries with indexes and lean projections, reducing content-editing time by ~20% and article load speed by ~30% on larger datasets.	

TECHNICAL SKILLS

Languages & Backend: C, Java, Python, JavaScript, TypeScript, SQL, Spring Boot, Flask, Express.js, REST APIs, Microservices, React, HTML/CSS, JWT Authentication
Data & Infrastructure: PostgreSQL, MongoDB, MySQL, Redis, Kafka, Spark, Kubernetes, Docker Compose, Unix/Linux, Nginx, Git, CI/CD, JUnit