

# Shane Edelstein

647-297-6393 | [shaneedelstein@gmail.com](mailto:shaneedelstein@gmail.com) | [Personal Website](#) | [LinkedIn](#) | [GitHub](#)

## EDUCATION

---

### University of Western Ontario

*Honors Specialization in Computer Science (BSc)*

London, ON

*Expected April 2027*

- Courses: Operating Systems, Computer Organization, Organization of Programming Languages, Theory of Computing, Object Oriented Design, Data Structures and Algorithms, Analysis of Algorithms, Intro to AI, Linear Algebra

## EXPERIENCE

---

### Brick works Academy

*Programming Instructor*

Jun 2025 – Sep 2025

*London, Canada*

- Taught fundamental programming concepts to students aged 9 to 13, covering core programming principles and logical thinking.
- Developed problem-solving skills by guiding students through structured solution design, breaking down complex problems into more manageable steps.
- Designed and delivered hands-on programming activities that enabled students to build, code, and troubleshoot their projects.

### FLL (FIRST LEGO League)

*Robotics Mentor*

Jun 2021 – Sep 2023

*Toronto, Canada*

- Mentored the award-winning Glen Ames P.S. Robotics Team; resulted in invitations to two international First Lego League robotics competitions.
- Instructed students in programming LEGO EV3 robots, covering key coding concepts and hands-on application of robotic functions.
- Delivered lessons on Python programming basics, emphasizing code structure, object-oriented code and the importance of clear documentation through comments when working with others.

## PROJECTS

---

### TorqueDesk | *C++, Qt5, Boost.Asio, SQLite*

- Built TorqueDesk, a full-stack automotive repair management application that helps customers and mechanics handle booking, communication, and repair tracking in one workflow.
- Led backend development in C++20 by building an asynchronous REST server with SQLite persistence, JWT authentication, role-based authorization, and transactional appointment-to-job state transitions.
- Implemented scalable concurrency using coroutine-based request handling and thread pooling to keep the server responsive while multiple users sent requests at the same time (for messaging, bookings, and job updates).
- Delivered the desktop client and quality workflow using Qt5, CMake/Ninja, and GoogleTest, with automated tests covering core service logic.

### Min-Max Life | *React Native, Typescript, Supabase, OpenAI*

- Developed a gamified productivity app using React Native (Expo), TypeScript, and Supabase, featuring user authentication, quest tracking, and real-time stat visualization. Leveraged React Query (TanStack Query) for efficient data fetching and caching, and Tamagui for cross-platform UI components supporting iOS, Android, and Web.
- Designed and implemented PostgreSQL database schema with 4 normalized tables and optimized queries using indexed columns. Developed RESTful API service layer with TypeScript for type-safe database operations including CRUD operations, stat tracking, and completion history.
- Integrated OpenAI GPT-4 API to automatically assign stat points (0-10) to user-created quests based on difficulty and activity type, reducing manual input.

### Farmers Market E-Commerce Website | *Next.js, PayloadCMS, MongoDB, Stripe Connect*

- Developed a reusable multitenant e-commerce template enabling users to create and manage their own digital product storefronts with isolated data and independent Stripe payment processing.
- Implemented Stripe Connect integration with separate Stripe accounts per tenant, platform fee handling, and webhook processing for automated order fulfillment and account verification.
- Developed a full-stack application using Next.js 15 and React 19, PayloadCMS with MongoDB, tRPC, and TanStack Query, featuring product catalog browsing, shopping cart, checkout flow, and a library for purchased products.
- Adapted template into specialized platform for farmers market with pickup-only ordering system.

## TECHNICAL SKILLS

---

**Programming Languages:** Java, Python, C, C++, SQL, JS/TS, HTML/CSS, ARM Assembly

**Frameworks/Libraries:** React, React Native, Node.js, OpenAI API, JUnit, Next.js, FastAPI, Jackson, Java-Swing

**Developer Tools:** Git, MongoDB, UNIX, VS Code, Figma, Lucidchart, Markdown, Latex

**Languages:** Native proficiency in English. Intermediate proficiency in French.