

Sean Behan

(781)-422-4938 • behan.s@northeastern.edu • Boston, MA • <https://github.com/swbehan>
<https://www.linkedin.com/in/sean-behan/> • [Personal website/ Portfolio](#)

EDUCATION

Northeastern University

Candidate for B.S in Computer Science

Concentration: Software Development

Minor: Business Administration

Relevant Coursework: Computer Systems, Grad Web Development, Grad Algorithms, Object-Oriented Design, Professional Practicum Capstone, Logic and Computation, Intro to Databases, Fundamentals of Computer Science 1 & 2, Discrete Structures, Intro to Math Reasoning, Tech and Human Values

Activities/Interests: Treasurer of NU Run Club (5k's, 8k's, Half Marathon, Marathon) and Intramural Flag Football

Honors: Dean's List (2024, 2025, 2026)

Boston, MA

May 2028

GPA: 3.95

TECHNICAL KNOWLEDGE

Languages: Java, Kotlin, SQL, C, Assembly, Python, CSS 3, HTML 5, Racket, Typescript, Javascript

Frameworks and Libraries: Flask, StreamLit, Pandas, Numpy, JUnit, Maven, Espresso, Jetpack Compose, Express.js, OkHttp

Technologies: GitHub, MySQL, Docker, IntelliJ, Android Studio, VS Code, DataGrip, Jira, CircleCI, Testmo, X-ray, MongoDB, Node.js, Claude Code, Gradle, BLE, Playwright, CI/CD

WORK EXPERIENCE

Bevi

Boston, MA

Software Engineer In Test Co-op

January 2026 - September 2026

- Developed a **CircleCI** job that builds a single APK and distributes it across various **Android emulators**, running unique test suites per branch — catching failures long before I executed formal regression testing
- Built multiple Android UI automation suites in **Kotlin** using **Espresso** with OOP principles and the **Page Object Model**, achieving full functional coverage and cutting days off regression testing
- Leveraged Jira, X-Ray and Testmo to author and maintain both manual and **automated** test cases, ensuring full coverage of functionality that cannot be automated while feeding documented test steps directly into the regression suite
- Built a **TypeScript**/Playwright e2e testing framework from scratch using the Page Object Model and **Claude Code Skills** to automate generation of new page objects and tests, establishing end-to-end coverage where none existed

Extended Co-op Special Project: **Bevi User Project**

May 2026 - August 2026

- Built an end-to-end user personalization feature for Bevi's on-machine Android app using Jetpack **Compose**, Kotlin **Coroutines/StateFlow**, and Kotlin **dependency injection**
- Implemented multiple phone-to-machine identity transports; signed **RFID payloads** (ECDSA / SHA-256), **Bluetooth Low Energy**, and QR-code-triggered **WebSocket** sign-in to create authenticated user sessions
- Built a **web app** and backend for managing personalized drink presets, loaded onto a Bevi machine by **QR scan** and applied only after the machine verifies a signed payload. Developed with **agentic engineering** workflows against codified project guidelines.

PROJECT EXPERIENCE

See [portfolio](#) for more projects

ClubSync: Co-Creator

May 2026- Present

- Built and shipped ClubSync, a **full-stack** club management platform (**React**, **Express**, native **MongoDB** driver) currently in active use by a Northeastern club, replacing manual spreadsheet/Google Forms workflows for dues tracking and event access
- Implemented session-based authentication (**Passport**, **bcrypt**, express-session backed by MongoDB) and a **tier-based eligibility system** where dues approval status automatically gates which events a member can RSVP to
- Designed **backend-paginated REST APIs** across multiple MongoDB collections without an ORM, handling 1,300+ records while keeping frontend queries **lightweight**

Rundown: Creator

March 2026- Present

- Built mobile training companion in **React Native** that integrates **Strava API** and **Claude API** to deliver real-time personalized coaching recommendations
- Built **verification framework** that auto validates running task completion by syncing Strava activity at 10pm, gating task unlock to **real workout data**, creating accountability through actual performance metrics
- Architected real-time sync between Strava **webhooks**, **Supabase**, and React Native frontend with **OAuth** authentication, maintaining responsive UX during LLM inference