

Jordan Glass

Hoboken, New Jersey 07030

jordanglass421@gmail.com | linkedin.com/in/jordan-glass-one | github.com/jordanglass21

SUMMARY

Full-stack software engineer with production experience shipping TypeScript/NestJS services, RESTful/OpenAPI-driven APIs, PostgreSQL, and automated testing, plus project experience building web apps end-to-end with React/Next.js.

EDUCATION

Northeastern University, Khoury College of Computer Sciences

Master of Science in Computer Science

August 2026

Boston, MA

Binghamton University - SUNY

B.A. Geography Computer Applications, Minor in CS

May 2019

Binghamton, NY

TECHNICAL SKILLS

- **Languages:** TypeScript, JavaScript, C#, Python, Java
- **Frameworks & Databases:** Node.js, Express, NestJS, React, Next.js, .NET, Spring Boot, PostgreSQL, MongoDB, Supabase
- **Testing, Cloud & DevOps:** Jest, Cypress, CodeQL, Git, GitHub Actions, CI/CD, Docker, REST/OpenAPI, Vercel

EXPERIENCE

Software Engineering Co-op

Hydrow

Sep. 2025 – Dec. 2025

Boston, MA

- Designed and built progress summary and calendar APIs in NestJS/TypeScript, generating dynamic stats and milestones across multiple time ranges and modalities (rowing, strength), with month-by-month navigation and streak tracking.
- Built five subscription authorization endpoints enabling owners to grant, revoke, and bulk-manage user permissions while enforcing rules like self-revocation limits and protecting admin access from accidental removal.
- Wrote Jest E2E and unit test suites covering every feature I shipped, validating business logic and edge cases before release.
- Contributed in an Agile environment via pair programming, code reviews, and sprint demos, presenting feature updates to the engineering team; maintained structured Git workflows tied to weekly CI/CD releases across Dockerized services.

Operations Manager

Star Wines and Liquors

Feb. 2020 – Aug. 2024

Monroe, NY

- Integrated third-party platforms (DoorDash, Vivino) and launched digital campaigns for an \$8M/yr retail business, generating over \$800K in annual online revenue.
- Managed daily operations, supervising a team of 20 while serving as primary point of contact for vendors and distributors on inventory, delivery, and contract terms.

PROJECTS

Shopping List | TypeScript, Next.js, React, Tailwind CSS, Supabase (PostgreSQL, RLS, OAuth), Vercel

- Built a full-stack web app that parses recipe ingredients into categorized shopping lists, auto-sorting items by supermarket aisle with real-time sync across devices.
- Enforced per-user data access at the database layer using PostgreSQL Row Level Security, allowing the client to safely query the database directly with a publishable key and no server-side authorization logic.
- Architected data logic into custom React hooks to keep components presentational, with schema changes version-controlled through Supabase CLI migrations and Google OAuth/magic-link authentication.

Full-Stack Q&A Platform | TypeScript, React, Node.js, Express, MongoDB, Jest, Cypress, GitHub Actions, OpenAPI, CodeQL

- Built a microservice-based Q&A platform with independently deployed services per resource, each persisting to its own MongoDB instance and validating requests/responses against OpenAPI specs via express-openapi-validator.
- Achieved 99% test coverage with Jest unit/integration tests, plus BDD-style Cypress E2E tests covering core user flows (posting questions/answers, voting, moderation).
- Automated test execution through GitHub Actions CI/CD pipelines, gating deployments on passing test suites.

AssetSync | C#, .NET 8, Java 21, Spring Boot, React, TypeScript, SQLite, Entity Framework Core

- Architected a three-service reconciliation platform combining a C# .NET 8 API, a Java Spring Boot mock legacy service, and a React/TypeScript dashboard, communicating over HTTP/JSON to test discrepancy detection against known, seeded data.
- Built a dictionary-backed O(1) comparison engine in EF Core/SQLite, applying decimal tolerance thresholds to distinguish real discrepancies from floating-point precision noise across warehouse sales, retail sales, and retail transfers.
- Developed a React/TypeScript dashboard to surface reconciliation reports, giving users visibility into item-level discrepancies between modern and legacy data sources.