

# Joshua Segal

[segal.jo@northeastern.edu](mailto:segal.jo@northeastern.edu) | [linkedin.com/in/joshua-francis-segal](https://linkedin.com/in/joshua-francis-segal) | [github.com/josh-segal](https://github.com/josh-segal) | [joshuasegal.dev](https://joshuasegal.dev)

## EDUCATION

### Northeastern University

Boston, MA

*Bachelor of Science in Computer Science, AI concentration, Math minor*

**Expected: May 2026**

**Honors:** GPA: 3.7 | Dean's List (all semesters)

**Relevant Coursework:** Algorithms | Object Oriented Programming | Artificial Intelligence | Machine Learning | Natural Language Processing | Neural Networks | Computer Systems | Database Design | Software Engineering

## TECHNICAL SKILLS

**Languages:** Python | TypeScript/JavaScript | SQL | Java | Rust | C

**Frameworks:** FastAPI | Flask | Django | React | Node.js | Express.js | SQLAlchemy | Prisma

**ML/AI:** PyTorch | TensorFlow | LLMs | RAG | Computer Vision

**Tools:** AWS (ECS, RDS, Lambda, S3) | PostgreSQL | Supabase | Docker | Terraform | Git

## EXPERIENCE

### Tech Lead, Software Engineer

July 2025 - Present

*NExT Consulting*

Boston, MA

- Led team of 6 engineers building production warehouse management system for hardtech manufacturer using **Python/React**, eliminating **20 hrs/wk** manual tracking, improving accuracy from **50%** to **95%** for **10k+** items
- Designed **PostgreSQL** schema with temporal row versioning and genealogy tracking for complete traceability across **100k+ rows**, and optimizing queries to achieve **2x** faster performance
- Architected and deployed **AWS infrastructure (ECS, RDS, S3)** using **Terraform** with **CI/CD** pipeline, supporting **250+** users

### Software Engineer

July 2024 - Dec. 2024

*Harvard Research Lab*

Boston, MA

- Built **Python** neuron segmentation desktop app reducing analysis time from **20 min to 0.5s**/image across **5000+** images, enabling **6+** researchers to label data, fine-tune **CNN** models, and run inference locally
- Designed local **NoSQL** database handling **100+** experiments with full persistence for reproducible training
- Increased segmentation precision from **61% to 93%** by implementing multi-user labeling using STAPLE algorithm

## PROJECTS

### Multi-Agent Lead Qualifier | [GitHub](#)

- Built multil-agent lead qualification system reducing response time from **2 hours to 4 seconds** across **1k+** messages/month using Python, Supabase, GPT-4o mini, Twilio webhooks
- Implemented custom agent router handling 5 agents with 99% routing accuracy using dynamic prompt injection
- Architected and deployed serverless AWS architecture (Lambda, S3) for **2 clients**, reducing costs by 90%

### Property Intelligence Platform

- Built property intelligence platform sourcing and analyzing **180k+** records using **FastAPI, React, PostgreSQL**
- Designed AI-powered web search saving **1k+ hrs/mo** research for **20+** users using **GPT-4o, Tavily API**
- Reduced integration time from **2 weeks to 2 days** using factory pattern for scrapers across **10+** locations

### Medical Claims Engine | [GitHub](#)

- Built actor-driven claims processing engine achieving **5k+ claims/second** using **Rust, Tokio**
- Designed lock-free actor coordination using **mpsc** channels, achieving **4x** speed improvement
- Implemented real-time aging reports with automated time-bucket aggregation and statistical analysis

### FUSE Filesystem

- Built filesystem achieving **90%** block utilization with concurrent access handling using **C, libfuse**
- Implemented free list allocation with coalescing and inode structures for hierarchical file organization