# Joshua Segal

860-938-5043 | segal.jo@northeastern.edu | linkedin.com/in/joshua-francis-segal | github.com/josh-segal

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

## Technical Skills

**Languages**: Python | Go | C | Rust | SQL | Bash
**Frameworks & Libraries**: FastAPI | PyTorch | TensorFlow | SQLAlchemy | Tokio
**Infrastructure & DevOps**: Linux/Unix | Docker | GCP | AWS | Terraform | CI/CD
**Databases & Tools**: PostgreSQL | MySQL | Git | Make | GDB | Valgrind

## Experience

**Tech Lead, Software Engineer** **July 2025 - Dec. 2025**
*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 **100k+ rows**, implemented alembic database migrations across dev/staging/prod, optimized data access patterns, achieving **2x** reduced network transfer volume
- Architected and deployed containerized **AWS infrastructure (ECS, RDS, S3)** using **Docker** and **Terraform** IaC with **CI/CD** pipeline (test, lint, build, deploy), supporting **250+** concurrent users

**Software Engineer** **July 2024 - Dec. 2024**
*Harvard Research Lab* *Boston, MA*

- Built end-to-end **ML infrastructure** training **U-Net** from scratch and fine-tuning **CNNs** locally, reduced analysis time from **20 min to 0.5s**/image; developed automated data pipeline handling preprocessing for **5000**+ images
- Designed local database schema managing model versioning, experiment tracking, and artifact storage for **100+** training runs with full reproducibility, enabled **6+** researchers to iterate on models locally
- Increased segmentation precision from **61% to 93%** by implementing multi-user labeling using STAPLE algorithm

## Projects

**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
- Debugged memory management and concurrency issues using GDB and Valgrind

**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

**Calendar Automation System** | GitHub

- Built **hybrid AI** system with locally-deployed fine-tuned **distilBERT** model for calendar classification, eliminating cloud API costs and reducing latency by **75%** across **200+** events
- Trained classification model using LLM-generated synthetic dataset of **10k+** examples to overcome data scarcity
- Architected Node.js inference server for local model serving with Notion API integration, saving **5+** hours/week

**Multi-Agent Lead Qualifier** | GitHub

- Built **multi-agent AI** 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 Lambda architecture (Lambda, S3) for **2** clients, reducing costs by **90%**