Benjamin Meyer

ben@bensmeyer.com | linkedin/benjaminsmeyer | github/benjaminsmeyer | bensmeyer.com

Education

Northeastern University Boston, MA
B.S. in Computer Science, Minor in Business Administration *Sept. 2022 – May 2026*
GPA: 3.9/4.0, *Summa Cum Laude*
Relevant Coursework: Software Engineering, Programming Languages, Artificial Intelligence, Machine Learning, Algorithms, Computer Systems, Large Language Models, Natural Language Processing
Teaching Assistant: Fundamentals of Computer Science I (Fall 2023) and II (Spring 2024)

Technical Skills

Languages: Python, TypeScript, JavaScript, Java, C#, SQL, C/C++, Kotlin, Go, Bash
Technologies: React, Redux, Node.js, Flask, ASP.NET Core, Spring Boot, FastAPI, GraphQL, REST APIs
Databases: PostgreSQL, MySQL, MongoDB, Redis, DynamoDB, Elasticsearch, Azure Cosmos DB
Infrastructure: Azure, AWS, Docker, Kubernetes, Git, GitHub Actions, CI/CD, Splunk, Datadog, Maven, Linux

Work Experience

Microsoft *July 2026 – Present*
Software Engineer Redmond, WA
- Returning full-time after internship.

Microsoft *May 2025 – Aug. 2025*
Software Engineer Intern Redmond, WA
- Architected and shipped a meeting insights feature for **Copilot for Sales**, integrating **GPT-4** and **Microsoft Graph APIs** to generate AI-powered meeting briefs directly within Outlook, eliminating pre-meeting research for sales teams.
- Built dynamic, accessible front-end experiences in **React** and **TypeScript** with **Redux** state management and **Fluent UI**, supporting enterprise-grade localization and WCAG compliance.
- Designed scalable backend APIs in **ASP.NET Core** and **C#**, and led deployment to **Azure Kubernetes Service (AKS)** with CI/CD pipelines via **Azure DevOps**, enabling zero-downtime releases to production.

Nara Logics *Jan. 2025 – May 2025*
Software Engineer Co-op Boston, MA
- Developed serverless APIs using **Python** and **FastAPI**, serving AI-driven recommendation systems at **1,000+ requests/minute** with **sub-100ms latency** and **98% test coverage**.
- Deployed ETL pipelines on **AWS** (EMR, S3, Lambda) processing **100GB+ daily** for ML workloads, and introduced **Redis** caching layers and parallel processing optimizations that reduced end-to-end processing time by **40%**.

HarbourVest Partners *Jan. 2024 – June 2024*
Security Engineer Co-op Boston, MA
- Built Python automation service integrating **Splunk** and **CrowdStrike** to triage and remediate security incidents, cutting response time from **4 hours to 15 minutes** and saving **300+ hours** of manual effort annually.
- Designed graph traversal algorithms to cross-reference **500M+ compromised credentials** from public breach data with company accounts, enabling proactive alerts and automated password reset workflows.

Projects

NYTimes Pips AI Solver | *Python, Flask, REST APIs, NumPy* *Sep 2025 – Dec 2025*
- Engineered AI agents leveraging CSP backtracking and Simulated Annealing to autonomously solve NYT Pips domino puzzles, achieving a **95%+** solve rate across all difficulty tiers in **sub-second** runtimes.
- Built a Flask REST API serving puzzle solutions as JSON, a GUI for real-time solve visualization, and a CLI for batch execution, benchmarking performance across **500+** puzzle configurations.

Kambaz Learning Management System | *Next.js, TypeScript, Node.js, MongoDB, Vercel* *Jan 2026 – Mar 2026*
- Developed a full-stack LMS in Next.js with server and client components, integrating MongoDB for persistent storage of **50+** courses, assignments, and **3 user roles** (student, faculty, admin).
- Implemented secure authentication flows, course dashboards, calendar, and inbox modules, achieving **90%+** test coverage and deploying to production on Vercel.

NLP Emotion Analysis | *Python, PyTorch, TensorFlow, Transformers* *Feb 2026 – Mar 2026*
- Trained and benchmarked four classification models (Naive Bayes, Logistic Regression, Bi-LSTM, and fine-tuned DistilBERT) on **20,000+** labeled tweets, with DistilBERT achieving **92% F1-score** across six emotion categories.
- Automated evaluation pipelines generating confusion matrices, training curves, and cross-model comparison charts, revealing transformer models outperformed traditional baselines by **15%+** on macro F1.