

Nick Trapp

Champaign, IL · ndtrapp2@illinois.edu · 630-277-7989

github.com/NickTrapp · nicktrapp.com · linkedin.com/in/nick-trapp

CS master's student at UIUC (graduating December 2026) seeking new grad software engineering and applied science roles. Interested in backend systems, machine learning, and optimization.

EDUCATION

University of Illinois Urbana-Champaign

Master of Computer Science · GPA 3.94 / 4.0

Aug 2025 – Dec 2026 (Expected)

B.S. Statistics & Computer Science · GPA 3.9 / 4.0

Aug 2021 – May 2025

Coursework: Machine Learning, Algorithms, Optimization, Reinforcement Learning, Algorithmic Game Theory, Database Systems, Systems Programming, Probability & Statistics.

EXPERIENCE

Microsoft — Software Engineering Intern · Redmond, WA

May 2026 – Aug 2026

- Building developer-facing APIs and a local MCP integration for Microsoft's Office licensing platform, helping engineers automate testing and troubleshoot end-to-end licensing workflows.
- Designed the integration so common licensing checks and workflow simulations can be run directly from developer tooling across environments.

Amazon Web Services — Software Development Engineering Intern · New York, NY

May 2023 – Aug 2023

- Migrated the FSx file-storage team's build and release pipelines to automated infrastructure using AWS CDK, reducing manual pipeline operations.
- Built a Java test package to run legacy pipeline scripts, executing Bash and Ruby to broaden coverage and enable a seamless migration.

Amazon — Software Development Engineering Intern · New York, NY

May 2022 – Aug 2022

- Built a full-stack internal tool (React + Ruby on Rails) surfacing action items from employee survey feedback; designed the database schema and APIs for scalable front-end / back-end integration.

PROJECTS

Combinatorial Auctions — Python, Gurobi

2026

- Engineered exact solvers for the Winner Determination Problem; custom branching heuristics ran up to 11× faster than Gurobi's default on hard mid-size instances, benchmarked across random and CATS families.

Approximate VCG Manipulation — Python, Gurobi, mechanism design

2026

- Showed time-limited VCG auction solvers open a profitable bid manipulation — overbidding 5× wins 72–75% of the time when the solver runs at 5–10% of exact solve time — and identified the branch-and-bound anchor effect behind the payment gap.

tv-writer-ai — Next.js, TypeScript, Prisma, PostgreSQL, Gemini

2026

- A workspace for writing a TV pilot with an LLM where output is a branch, not an answer; promoting a branch folds it into project canon that constrains later generation to keep continuity.

poker-coach — Python, LLMs, pytest

2026

- An AI coaching system where *the model explains but never calculates*: equity, pot odds, EV and MDF come from a deterministic layer, and a checker fails any response that cites a figure it wasn't given. 872 tests.

RAG Document QA — Python, ChromaDB, Gemini

2025

- Retrieval-augmented QA over private PDFs, with OCR fallback for scanned documents, sentence-boundary chunking, and a ROUGE-scored evaluation set.

SKILLS

Languages: Python, C++, Java, Kotlin, TypeScript, JavaScript, SQL, C, R

ML & Data: PyTorch, NumPy, pandas, scikit-learn, Gurobi

Web & Backend: React, Next.js, Node.js, Flask, Ruby on Rails

Infrastructure: AWS, AWS CDK, Docker, Google Cloud, Git