Cyrus Singer

 github.com   linkedin   **cyrus.singer.dev**   cysinger@seas.upenn.edu   (561) 403-8133

🎓 EDUCATION

University of Pennsylvania	May 2025
Bachelor of Science And Engineering in Computer Science	*Current GPA: 3.57/4.0*

</> SKILLS

Languages: JavaScript/TypeScript, Java, Kotlin, Python, C/C++/C#, Haskell, Ocaml, SQL, HTML, CSS
Software Tools: MATLAB, GCP, AWS, Firebase, Docker, Git, GDB, Selenium
Software Frameworks/Libraries: React, Node.js, Express.js, Pytorch, Tensorflow, pandas, NumPy, tRPC, Prisma
General Skills: embedded-programming, full-stack development, system-level programming, customer engagement, source control, CI/CD, product management, version control, linux, powershell, bash, REST, HTTP

💼 WORK EXPERIENCE

Software Engineering Lead on Bizzybots Platform | *Wharton Behavioral Lab*   2022 - Present
- Lead technical, architectural design and development of a web platform for creating custom, highly reliable chatbots used by over 500 students and professors across 5 universities for education and research (cited by papers under review)
- Oversee a team of five developers, enforcing coding, version control, logging best practices
- Design the platform using **OpenAI** LLMs for response generation and bot control, **React.js** for the frontend and **Express** for the backend, with **Firestore** as the database and **Firebase** user management and authentication
- Orchestrate deployment on **GCP**, using **Docker** and **Cloud Run**, ensuring **99.9% uptime**
- Design a system of independent **co-existing software versions**, allowing for rapid deployment of new features
- Implement a comprehensive **Jest** and **Selenium**-based test suite, **reducing production bugs by 80%**
- Conduct infrastructure load testing to verify the architecture can handle **10,000 messages per second**
- Publish and maintain an Excel-style formulas package link and a number parsing package link on **npm**

Reference and demo available upon request

Software Developer Intern | *CFO4ALL at Penn Venture Lab*   May 2024 - Sep 2024
- Worked on a web application to provide CPG companies with cash flow forecasting and insights:
- Set up **CI/CD** pipeline using **GitHub Actions** & **Docker** to deploy to **AWS App Runner** & **CloudFront**; reducing deployment time by 90% and allowing production credentials to be removed from developers' computers
- Designed a scalable time series-tree data structure for financial data; integrated this data structure with backend analysis and frontend visualizations; **speeding up queries by 70%**
- Implemented a data ingest pipeline from Quickbooks to a **PostgreSQL** database on **Heroku** for customers with **250,000+ data points**; **automating the customer onboarding process**

Teaching Assistant for Internet and Web Systems | *University of Pennsylvania*   Jan 2024 - Present
(MS/PhD Required Class)
- Guide students to debug code and understand course material through weekly office hours and an online forum
- Mentor groups through the final project, a distributed search engine with **1,000,000** indexed pages

💻 PERSONAL PROJECTS

Webapp to play chess with an LLM | *TypeScript, Next.js, LLMs, Minimax,* | GitHub   2024
- A web application that allows users to play chess against a chess engine powered by LLMs
- Utilizes a compressed, budgeted, form of the **Minimax algorithm**, with LLMs generating moves and evaluating game states, running in parallel for scalable performance
- Included logging and tuning capabilities, allowing for easy monitoring and adjustment of the algorithm's parameters

AI Writing React At Runtime | *TypeScript, React, esbuild, express.js* | GitHub   2024
- Developed an innovative solution enabling AI to generate and integrate a React component into a live site **without a page refresh**
- Evaluated different AI models on different platforms to balance generation quality and latency
- Constructed a TypeScript to CommonJS pipeline to reduce runtime failures of AI components by 90%

Brittle Object Simulation | *Python, GPU optimization, Physics Simulation* | GitHub   2022
- Created a program that simulated the internal stresses of rigid lattices under arbitrary forces and collisions using **Newtonian mechanics**
- Estimated internal bond stresses using **gradient descent** to fit internal stresses to the lattice's Newtonian motion
- Optimized the **bond estimation algorithm** to run with fewer gradient descent iterations
- Reimplemented the algorithm to run on a GPU, **improving performance over 10x**