Thomas Zeng

thomaszeng05@gmail.com https://www.linkedin.com/in/thomasz5 https://github.com/thomasz5 https://thomasz05.me

Education

University of Washington **Graduating: June 2027**

B.S. in Applied & Computational Mathematical Sciences and Informatics — Deans list *Seattle, WA*

Coursework: Data Structures & Algorithms, Discrete Mathematics, Computer Systems & Assembly, Operating Systems, Software Design & Implementation, Database Systems, Machine Learning, Web Development

Skills

Coding Languages: Java, Python, JavaScript, TypeScript SQL, MATLAB
Frameworks and Libraries: Flask, React.js, TailwindCSS, WebSocket libraries, D3.js, OpenCV, TesseractOCR, PyTorch, HuggingFaceTransformers, RasaNLU, pytest, Click, Sphinx, Docker, AzureSDK, Redis, Supabase client libraries, pysnow

Experience

CatanAI, Inc. **May 2025 – Present**

Founder & CTO *Remote*

- Founded and building AI-powered Catan strategy platform serving 20 players, with Flask/PostgreSQL backend, React/TypeScript frontend, real-time WebSocket infrastructure and OpenAi chatbot for strategy coaching.
- Engineered computer vision pipeline using OpenCV and Tesseract OCR to automatically detect board states from screenshots with 95% accuracy while processing 50+ board configurations daily.
- Established partnership with content creators with 10,000+ subscribers, enabling seamless integration with online Catan platforms

Pncel – University of Washington **Dec. 2024 – Apr. 2025**

Undergraduate Research Developer *Seattle, WA*

- Built Domain-Optimized Reconfigurable Array Framework core in Python, automated docs generation with Sphinx to produce an HTML reference site for researchers, reducing onboarding time by 3 hours .
- Formulated a CLI regression-testing tool using Click and pytest to validate end-to-end builder workflows

Tri Counties Bank **June 2024 – Sept. 2024**

Project Management Intern *Sacramento, CA*

- Configured NetBox project in Azure using Python with the Azure SDK and Requests library; built a Flask microservice for real-time Grafana dashboards and automated anomaly alerts, reducing manual audit time by 90%.
- Automated ServiceNow environment workflows with the pysnow library and ldap3 for AD/SSO sync; deprovisioned 73 dormant accounts and reclaimed unused licenses

iCare – University of Washington **Dec. 2023 – May 2024**

Undergraduate Machine Learning Researcher *Seattle, WA*

- Implementated RASA NLU model and dialogue management system to handle complex user queries, using Python to generate personalized responses and increasing accuracy of user interactions from previous implementations.
- Finetuned most adaptable personality profile summarization model, implementing end-to-end testing to confirm fit between margins by nearly **98%** using long term weighted averaging.

Algorithmic Trading Club – University of Washington **Sept. 2023 – Dec. 2024**

Co-President *Seattle, WA*

- Organized and developed Husky Hold'em Coding Competition and Website via React.js., Tailwind, securing **$6500** worth of sponsorship support, hosting **3** keynote speeches, **50+** participants and **10+** mentors.

Projects

Dr. Mike AI Fitness Chatbot — Python, Pytorch, Redis, Docker, HuggingFace, React, Supabase, PostgreSQL

- Developed a personality-driven AI chatbot to mimic fitness expert's knowledge, enhancing user interactions by 30%.
- Processed and scraped 15+ hours of video transcripts using Python scripting and OpenAI API to extract personality/knowledge.

Autonomous Poker Bot — Python, SQL, JavaScript, PyTorch, PostgreSQL, Redis, Docker

- Built poker AI system that achieved 2.1 BB/100 win rate through multi-stage machine learning pipeline combining imitation/reinforcement learning, and GTO strategies
- Established testing framework with 400+ unit tests and achieving code coverage and validation across 10M+ test scenarios

Recruitment Intelligence System — Python, SQL, Pandas, AWS Ec2, Selenium, BeautifulSoup

- Developed Python-based web scraping system for internships and processing 100+ opportunities weekly
- Deployed production system on AWS EC2 with 3 hr monitoring cycles, reducing manual job search time by 50%