Elijah Baraw

ebaraw@andrew.cmu.edu | (203) 731-9535 | Pittsburgh, PA | github.com/elijah-bae-raw

EDUCATION

Carnegie Mellon University, School of Computer Science   **Aug 2021 – May 2025**
Bachelor of Science in Computer Science. Concentration in Computer Systems
GPA: 3.97. Relevant courses: Machine Learning, Cloud Computing, Distributed Systems, Data Structures and Algorithms, Functional Programing, Systems, Parallel Algorithms, Linear Algebra, Differential Equations

TECHNICAL SKILLS

Languages: C, Python, Go, SQL, Java, HCL
Technologies: NumPy, PyTorch, Pandas, OpenCV, Linux, Sockets, Git, AWS, GCP, Azure, K8s, Docker
Topics: Data Structures and Algorithms, Object Oriented Programing, Functional Programming, Systems, Consensus Algorithms, Actor Model, Network Protocols, TCP, Cryptographic Algorithms, Machine Learning

EXPERIENCE

Center for Atmospheric Particle Studies   **Pittsburgh, PA**
Research Assistant   **May 2022 – Aug 2022**
- Developed a low cost device for measuring PM2.5 air pollutants collected on a foam tape over several months
- Implemented an image processing pipeline as a cheaper alternative to traditional particle detection machines
- Integrated a system of Arduino and Python scripts to control stepper motors based on continuous CV input.

PROJECTS

VirtualPup (Python Hackathon) *Selenium, NTLK, Tkinter, MVC architecture*   **Mar 2022**
- Designed an interactive game in 24 hours. Users customize a virtual dog, then chat with their pet with a textbox.
- Used NLTK for sentiment analysis and Selenium to allow the dog to "fetch" images from the web to play with
- Designed the game in 24-hours in a team of 3, submitted to a panel of judges, won first out of 25+ teams.

Fontify (Solo Python Project) *PIL, Image Processing, De-Noising, Computer Vision*   **Jan 2022 – July 2022**
- Created an image processing software in Python to convert handwritten letters into a personalized bitmap font.
- Utilized image processing, edge-detection, noise reduction algorithms to detect pencil writing on paper.

Concurrent Proxy Server (C) *Git, HTTP, Sockets*   **July 2023**
- Developed a proxy server in C using p_threads and fork to handle requests concurrently, anonymize traffic and cache responses. Utilized Unix sockets and a bounded cache following an LRU eviction policy.

Distributed Backend (Golang) *Replication, Actor Model, Mailbox/Message Passing*   **Nov 2023**
- Designed and executed a concurrent server to manage the state for a multiplayer game, accessible via API.
- Handled client requests about and updates to the game state using RPCs and a message-passing model.
- Implemented node launching and server groups, ensuring replication and enforcing consistency within groups.

Poker-Bots Hackathon Dev Team *GCP, K8s, GitHub Actions*   **Mar 2024**
- Helped CMU Data Science Club run their first AI Poker bot competition, with $6,000 in prizes and 63 teams.
- Used GitHub actions to automatically build docker images of user-submitted Python bots, allowing competitors to use custom dependencies and machine learning libraries of their choice.
- Deployed containers on GCP to run matches between submitted bots and monitored their status using kubectl.