Allen Cao

allencao@berkeley.edu (510) 935-3270
https://allencaoo.github.io https://www.linkedin.com/in/allen-l-cao https://github.com/AllenCaoo

EDUCATION

University of California, Berkeley **Berkeley, CA**
B.A. Computer Science, GPA: 3.86/4.0 *August 2020 - May 2024*

- *Coursework*: Data Structures, Efficient Algorithms, Computer Architecture, Database Systems, Operating Systems, Artificial Intelligence, Principles of Data Science, Computer Security, Information Devices (circuits), Linear Algebra, Discrete Mathematics and Probability

SKILLS

Programming Languages: Python, Java, C/C++, C#, TypeScript/JavaScript, HTML/CSS, Rust, SQL, Scala, Bash, Go

Frameworks: Node.js, Express, React, Django, Flask, Spring, Ruby on Rails, GraphQL, Apache Spark, MySQL, PostgreSQL, .NET, OpenCV

Tools: AWS (Lambda, Cloudformation, DynamoDB, Athena/Glue, S3, API Gateway, Redis, RDS), Git, MongoDB, Unity, Docker, Kubernetes

WORK EXPERIENCE

Amazon Fashion **Seattle, WA**
Software Development Engineer Intern *May 2023 - Present*

- Reduced week-long data management tasks to mere seconds by building a highly efficient content database administrator tool to enable seamless data search, modification, and migration.
- Achieved a 97% decrease in data retrieval latency by designing and implementing a downstream database partitioning scheme with Apache Spark, resulting in fast and predictable query latencies.
- Developed a scalable REST API using API Gateway, integrated with a React frontend and Lambda, Athena/Glue backend to serve thousands of Amazon Fashion content managers and data engineers daily.
- Constructed deployable cloud infrastructure using AWS CloudFormation/CDK and integrated across 3 development environments.

Baxter International **Deerfield, IL**
Software Development Engineer Intern *June 2022 - August 2022*

- Fully obtained FDA compliance by incorporating an interrupt service routine into medical device systems to monitor device damage, collect data, and generate logs to an end-user database.
- Upgraded internal API by developing a logging queue and integrating throughout codebase to resolve thread-safety concerns and optimize multi-threading.
- Constructed a streamlined Python machine learning workflow that includes an ETL pipeline feeding into a damage detection classifier, resulting in a model with 99% accuracy on test data.
- Automated internal GUI quality control by building an OpenCV application to identify missing design requirements from GUI images.

UC Berkeley EECS, Barsky Lab **Berkeley, CA**
Computer Vision Researcher *August 2022 - Present*

- Created Unity plugins to compile and integrate vision filtering algorithms into dynamic link libraries for Android and IOS app development.
- Developing a network-based research workflow to enable cross-language and cross-platform communication for C/C++ algorithm experimentation in .NET/C# apps.

Leopard Imaging **Fremont, CA**
Software Engineering Intern *May 2021 - August 2021*

- Accelerated the examination and analysis of circuit failures by developing a defect simulation tool that empowers engineers to manipulate circuit images and inexpensively replicate realistic faults.
- Created an OpenCV image filter tool that rapidly identifies faulty circuit components with over 99% correctness.

PROJECTS

Pathfind Visualizer *React*
https://allencaoo.github.io/Pathfind-Visualizer

- Built an interactive visualization tool for pathfind and maze generation algorithms on a 2D grid.
- As course staff, released the project to over 500 students in UC Berkeley's data structures (CS61BL) course as a visual learning tool.
- Algorithms include BFS, DFS, Dijkstra's, A*, Greedy Best First Search, Randomized Prim's, and Inverted Randomized Prim's.

tAI *MongoDB, Node, Express, React, Redux*

- Built an AI-powered educational tool delivering personalized assistance to all educational levels and subjects and generating reports to educators.
- Features include feedback-based quizzes, conversational office hours, and grading interface.
- Utilized MongoDB to manage content and student feedback, Express.js and AI text generation APIs for backend, and React for UIs.

Pintos OS *C, x86 Assembly*

- Programmed a uniprocessor operating system to handle and manage interrupts, system calls, memory allocation, thread scheduling, extensible file system (Berkeley FFS), caching, and virtual memory.
- Implemented comprehensive thread and process synchronization using locks, semaphores, and monitors.

Gitlet *Java*

- Designed and implemented a version control system for local and remote repositories.
- Built an organized hash-based storage system for commits, branches, and remote repositories.
- Supports init, add, commit, rm, log, global-log, status, checkout, branch, rm-branch, reset, merge, add-remote, rm-remote, fetch, push, pull.