Paul Struble

+1 (650)-554-9895 | paul_struble@berkeley.edu | paulstruble.dev | linkedin.com/in/paulstruble

Education

University of California Berkeley Graduating Spring 2026
B.A. Computer Science & B.A. Data Science *Cumulative GPA: 3.75*

Relevant Coursework: Data Structures, Advanced Algorithms, Machine Learning, Database Systems, Computer Architecture, Computer Security, Internet Architecture, Artificial Intelligence, Principles of Data Science, Probability, Discrete Mathematics, Linear Algebra & Differential Equations, Multivariable Calculus

Experience

Guepard December 2024 - Present
Software Engineer Intern (Systems)
- Reduced server storage requirements by up to 3x for development teams by tracking database diffs in a virtual layer, eliminating storage duplication and bloat
- Accelerated feature development (for customers) by up to 64% and increased branches per project by up to 6x by providing database-agnostic version control features in Guepard's virtual layer
- Utilized database- and systems-design principles while contributing to Guepard's Rust codebase
- *Key Technologies: Rust, Systems Programming, System Design (Databases, Virtual Machines, Filesystems, etc.)*

bWork Live October 2024 - December 2024
Software Engineer Intern (Full Stack)
- Built a real-time interactive dashboard webapp for UC Berkeley Housing to monitor and visualize maintenance data for more than 7,000 residents across 54 University properties
- Developed a REST API with Express to efficiently fetch 1000's of records from a 1.1M-record database
- Automated PostgreSQL database updates to handle 300+ daily maintenance orders/requests by using a web scraper
- *Key Technologies: TypeScript, React, Next.js, Node.js, Express.js, Tailwind, PostgreSQL, Docker, AWS*

RookieDB September 2024 - November 2024
Systems Software Engineering
- Built an ACID-compliant relational DBMS by adding key components/algorithms to a large-scale OOP Java codebase
- Achieved exponential runtime improvements by implementing core features including: B+ Tree indexes, 6x unique join algorithms, a 3-layer multi-granularity locking system (for transaction concurrency)
- Enforced data durability for 0% data loss upon system recovery by implementing the ARIES recovery algorithm
- *Key Technologies: Java, JUnit, Git, Database Design, System Design*

bScraper September 2023 - June 2024
Software Engineering (Python)
- Developed a web scraper that collected 16+ years of UC Berkeley maintenance using Selenium and BeautifulSoup4
- Extracted historical maintenance data to a PostgreSQL database, including 750,000+ work orders and 400,000+ maintenance requests, allowing for analysis and use in future applications
- Increased scraper throughput to achieve 10x efficiency gains by parallelizing operations
- *Key Technologies: Python, PostgreSQL, Selenium, BeautifulSoup4*

Projects

Multithreaded Web Server | *Rust, TCP, HTTP, Systems Programming*
- A multithreaded HTTP web server built in Rust from scratch. Used only Rust's standard library to implement the server and facilitate communication over TCP, taking advantage of Rust's parallelization features for increased efficiency.

Convolution Optimization | *C, Parallelization, Optimization, Systems Programming*
- A low-level optimization of the convolution operation (e.g. matmul) in C using various parallelization and optimization techniques such as SIMD vector operations (x64 AVX), multithreading, loop unrolling, algorithmic optimization, etc.

Technical Skills

Languages: Python, Java, Rust, SQL, TypeScript, JavaScript, HTML/CSS, C/C++, RISC-V Assembly, Lua, Scheme

Technologies: Git, Docker, Linux, Node, Express, React, Next, Tailwind CSS, Postman, PostgreSQL, MySQL, SQLite, SQLAlchemy, MongoDB, Apache Spark, AWS, Rest APIs, Web Scraping, NumPy, SciPy, SymPy, Pandas, Matplotlib, PyTorch, scikit-learn, gdb, pytest, JUnit, Vim, VS Code, IntelliJ, Adobe Creative Cloud, Logisim Evolution