Rian Puri

rianpuri01@gmail.com | linkedin.com/in/rianpuri | github.com/rpuri4 | rianpuri.github.io

Education

University of California, Berkeley — Berkeley, CA
B.S. Electrical Engineering & Computer Science (Honors); GPA: 3.7 — *Aug. 2022 – May 2025*

Experience

Delivr AI — May 2025 – Aug 2025
Software Engineering Intern — *Satellite Beach, FL*

- Built a RAG-based microservice in Python to classify web traffic into an 87K-topic taxonomy, exposing fit scores and reasoning via REST APIs.
- Developed low-latency Snowflake pipelines (SQL + Python) for real-time analytics and integration with audience targeting systems.
- Automated ingestion and processing with Docker + AWS Batch/Lambda, boosting throughput by 20% and eliminating manual ops.
- Added observability using OpenTelemetry for end-to-end latency and error tracing in production.

RecVue — June 2024 – Aug 2024
Software Engineer Intern — *Palo Alto, CA*

- Designed a FastAPI backend converting natural-language queries into validated SQL via LLM function-calling, enabling analytics access for non-technical users.
- Integrated PostgreSQL (RDS) + pgvector + Redis for semantic schema search and sub-100ms caching.
- Automated CI/CD with GitHub Actions + AWS ECS, improving deployment speed and observability using distributed tracing.

Algorithms for Computing and Education Lab (ACE), UC Berkeley — Jan 2025 – Present
Undergraduate Researcher — *Berkeley, CA*

- Building backend infrastructure for AutoRemind, a Flask-based system integrating LMS APIs to deliver adaptive learning reminders.
- Developing PostgreSQL data pipelines mapping assignments to relevant readings and learning resources.
- Designing analytics triggers detecting incomplete tasks and delivering personalized nudges, improving engagement.

Projects

AI Agent Orchestrator | *Go, Redis, gRPC, Docker, Kubernetes*

- Developed a multi-agent orchestration framework supporting concurrent agent execution, streaming, and pluggable LLM tools.
- Implemented task scheduling, state persistence (Redis), and gRPC interfaces for scalable coordination.
- Added Prometheus metrics and structured logging for observability, with fault recovery via retry queues.

Distributed Log Storage Engine | *C++, Raft, Linux, Perf*

- Designed a replicated, log-structured storage engine with append-only semantics, segment compaction, and replication.
- Implemented Raft-style consensus for leader election and fault tolerance across nodes.
- Optimized I/O and batching, achieving 70K ops/sec and <3ms median latency.

OCaml Tensor Compiler | *OCaml, Compiler Design, OpenBLAS, Autodiff*

- Extended a CS164 Lisp compiler with a Tensor type system, shape inference, and OpenBLAS bindings for high-performance matrix ops.
- Added reverse-mode autodiff, bytecode fusion, and peephole optimizations for constant folding and broadcast simplification.
- Implemented an arena allocator and optimized GC, reducing pause times by ∼35% on compute-heavy workloads.

Technical Skills

Areas: Backend Development, Distributed Systems, Databases, Infrastructure, Cloud, Security
Languages: Go, C++, Python, Java, OCaml, SQL, JavaScript/TypeScript, Bash, C
Tools: Docker, Kubernetes, AWS, gRPC, FastAPI, PostgreSQL, Redis, Snowflake, Kafka, GitHub Actions, Linux