JUHUN PARK

202-924-4546 | juhunpark32@gmail.com | juhunpark.me | linkedin.com/in/juhun-park | github.com/juhun32

EDUCATION

George Mason University Expected May 2027
Bachelor of Science in Computer Science GPA: 3.9

Relevant Coursework: Data Structures & Algorithms, Low-level Programming
Honors: $20,000 Merit Scholarship Recipient

EXPERIENCE

Google Developer Groups Aug 2025 - Present
Technical Lead, George Mason University Fairfax, VA

- Architected a course scheduler with Go and PostgreSQL to ingest and manage 10,000+ university course sections.
- Engineered a data pipeline using Python and GitHub Actions to automatically scrape course catalog with 99.9% accuracy.
- Integrated backtracking algorithm to solve the course scheduling problem, generating conflict free timetables in sub-2s.
- Scaled with RabbitMQ and Redis for asynchronous processing, supporting 100+ concurrent users without degradation.

Northwestern University Jun 2025 - Aug 2025
Software Engineer Intern Evanston, IL

- Built an alumni networking platform with SvelteKit and ASP.NET, serving 150 users with 99.9% uptime on Cloudflare.
- Engineered a PostgreSQL schema, optimized indexes SQL functions on Supabase achieving p99 sub-5ms query latency.
- Implemented a multi-tier caching strategy using Cloudflare CDN and a Redis cache, offloading 95% of read requests.

PROJECTS

Copium.dev | GitHub | copium.dev
Go, TypeScript, SvelteKit, PostgreSQL, Algolia, GCP, AWS, Cloudflare

- Built an internship management platform with SvelteKit and PostgreSQL, processing 2000+ weekly applications.
- Delivered an 80% reduction in query latency for BigQuery analytics by leveraging CQRS architecture.
- Set up an object storage pipeline using Cloudflare R2 and Go, offloading S3 and eliminating 100% of AWS egress costs.
- Implemented compensating transactions for consistency across 3 data stores with a 99.9% successful retry rate.

Californian Temple | GitHub | calple.date
Go, Next.js, Firestore, Docker, GCP Cloud Run

- Implemented an event pipeline using GCP Pub/Sub to ingest disparate streams into a unified Firestore state model.
- Developed a Go microservice to consume event streams, merging data real-time to predict and identify schedule conflicts.
- Achieved sub-100ms latency for event synchronization by optimizing Firestore listener patterns & gRPC communication.

Sequantial | GitHub
Python, Kafka, gRPC, Docker, Git, AC/acsys (Assetto Corsa Internal API)

- Built a Python data pipeline to ingest real-time telemetry from Assetto Corsa, interfacing directly with the memory API.
- Streamed 10k+ events/s to a Kafka topic, enabling decoupled real-time analytics for live Next.js dashboards.
- Utilized Kafka's partitioning and consumer groups to ensure zero data loss during connection loss, for fault tolerance.

TECHNICAL SKILLS

- **Languages:** Python, TypeScript, JavaScript, Go, Java, SQL, C, C#
- **Frameworks:** React, Node.js, Django, Flask, Next.js, SvelteKit, FastAPI, .NET, PostgreSQL
- **Tools & Platforms:** Git, GCP, Vercel, Cloudflare, Pub/Sub, AWS, Linux, RabbitMQ, Docker, Redis, Kafka