Farhan Mir

Easton, PA | (951) 732-7325 | farhansmir26@gmail.com | linkedin.com/in/farhan-mir | github.com/farhanmir | farhanmir.surf

Education

Rutgers University New Brunswick, NJ
Bachelor of Science in Computer Science *Expected May 2028*
GPA: 3.75 | Dean's List

Relevant Coursework: Data Structures & Algorithms, Database Management Systems, Software Design & Methodology, Data Science & Analytics, Computer Architecture, Discrete Mathematics

Experience

Software Engineering Intern (Backend) June 2026 – August 2026
Impiricus *Atlanta, GA*

- Engineered high-performance backend services for a healthcare platform serving the largest opted-in network of Healthcare Professionals; utilized Python and Go to enable real-time clinical insights for millions of users.
- Accelerated patient access to treatments by architecting PostgreSQL data pipelines and LLM integrations that delivered automated clinical resources with sub-second latency at enterprise scale.
- Optimized cloud infrastructure on AWS to support multi-tenant AI features; implemented advanced prompt engineering and data modeling to ensure high-accuracy, real-time medical resource delivery.

Software Engineering Intern Feb 2026 – May 2026
Eudaimonic Inc. *East Setauket, NY (Remote)*

- Boosted developer productivity by 65% by reducing widget compilation times from 180s to 60s; engineered a high-throughput engine using concurrent esbuild batching and adaptive concurrency control.
- Optimized financial data processing, reducing billing query latency by 70% and database memory overhead by 40MB per request through the implementation of composite indices and optimized SQL lookups.
- Scaled real-time collaborative editing to 30+ concurrent users while cutting storage writes by 88%; built an intelligent state-management layer that maintained sub-50ms synchronization latency.

Founding Engineer Jan 2026 – Present
ScarletSpots *New Brunswick, NJ*

- Architected a real-time monitoring platform sustaining 1,000+ concurrent connections with sub-50ms latency; engineered a high-throughput WebSocket transport layer using Redis pub/sub to handle massive traffic spikes.
- Achieved 94% prediction accuracy for occupancy forecasting by developing a hybrid inference model that toggles between pattern-based and observed data across four distinct time horizons.
- Eliminated 100% of data race conditions during peak usage periods; implemented atomic transaction patterns and serializable isolation in SQLAlchemy to ensure data integrity under high concurrency.

Projects

Lazarus | *Python, FastAPI, React, Neo4j, Three.js, Dedalus*

- Awarded the Regeneron Track Prize at HackPrinceton for architecting an autonomous clinical R&D swarm; orchestrated distributed multi-agent workflows using Dedalus Containers for end-to-end medical research.
- Reduced medical research time from hours to minutes by deploying Gemini models across a parallelized agent network, enabling high-speed synthesis of complex clinical data and literature.
- Ensured 100% data isolation during high-throughput inference; engineered a fault-tolerant orchestration pipeline to manage independent agent memory with zero cross-contamination.

Seamless | *Python, Snowflake, React, MCP, Playwright*

- Built a full-stack recommendation engine achieving sub-50ms overlay rendering; engineered a structured ingestion pipeline that processes 10,000+ real-time product mentions into Snowflake.
- Improved user discovery speeds by building a custom Model Context Protocol (MCP) server; integrated live web scraping with a React frontend to deliver targeted product queries in under 2 seconds.
- Reduced checkout latency by 60% and eliminated redundant scraping operations; designed an intelligent in-memory caching system to persist user state and improve overall conversion rates.

Merlin | *Next.js, FastAPI, Docker, PostgreSQL*

- Improved AI reasoning accuracy by 40% by engineering a proxy server implementing Monte Carlo Tree Search (MCTS) and Self-Consistency decoding for high-quality response generation.
- Architected a concurrent multi-model chat interface utilizing Server-Sent Events (SSE) to stream real-time reasoning steps from multiple AI providers simultaneously.
- Reduced compute costs by 35% by building a dynamic routing layer that selects models based on calculated prompt complexity and token entropy to balance cost and performance.

Technical Skills

Languages: Python, Go, Swift, TypeScript/JavaScript, C/C++, Java, SQL, Bash
Frameworks & Libraries: FastAPI, Swift UI, React, Next.js, React Native, Node.js, SQLAlchemy, Swift Concurrency, Pandas
Databases & Cloud: PostgreSQL, Snowflake, Redis, Supabase, MySQL, PostGIS, AWS (EC2, S3), Oracle Cloud (OCI), Vercel
Tools & Infra: Docker, Kubernetes, GitHub Actions, Linux (Ubuntu), Pytest, XCTest, SwiftLint