

# Mohammad Pervaiz

[momoperv123@gmail.com](mailto:momoperv123@gmail.com) | [linkedin.com/in/mohammad-pervaiz](https://linkedin.com/in/mohammad-pervaiz) | [github.com/momoperv123](https://github.com/momoperv123)

## EDUCATION

### Queens College, City University of New York

Bachelor's in Computer Science

May 2027

GPA: 3.8 / 4.0

**Leadership:** Former President, Code for All (2K+ members); IAC Fellow (1 of 20 nationwide; \$20K grant for an NPO)

## EXPERIENCE

### Bloomberg LP

Software Engineering Intern (*Returning*)

June 2026 – August 2026

New York, NY

### Bloomberg LP

Software Engineering Intern

June 2025 – August 2025

New York, NY

- Built a centralized and scalable Kafka-based CDC pipeline for 26K+ employees, replacing inefficient batch jobs
- Delivered IAM-guarded updates from 5+ upstream services by implementing batching in a distributed system
- Engineered payload delivery guarantees via Kafka transactions to 3+ ETL consumers using exactly-once (EOS)
- Achieved monitoring coverage for poller error paths by adding alerts via dashboards for logging and telemetry
- Developed 150+ automated unit, integration, and system tests within Docker, securing ~90% code coverage

### Morgan Stanley

Software Engineering Intern

December 2024 – January 2025

New York, NY

- Expanded coverage for 10+ JSON fields in a validation REST API with schema checks before deprecation
- Prevented potential outages by blocking invalid payloads, ensuring data integrity for dependent trading systems
- Wrote unit tests for 20+ API endpoints and collaborated with compliance to meet regulatory requirements

### People Inc

Software Engineering Intern

June 2024 – August 2024

New York, NY

- Shipped user-facing features for People.com (8M+ daily visits) and Entertainment Weekly (1M+ daily visits)
- Scaled site functionality with 270M+ monthly viewers while via profiling ad-serving paths to avoid regressions
- Standardized development across 40+ brands by integrating reusable components into People Inc's monorepo

## PROJECTS

### Multiplayer Game Server | C++17, CMake, Winsock2, TCP Sockets, MinGW-w64

[GitHub](#)

- Built a TCP multiplayer game server with client threads, an authoritative state, real-time chat, votes, and roles
- Designed a packet protocol for chat and votes with a thread-safe queue, socket I/O, and round-based updates
- Set up CMake builds on Windows via Winsock2, clean shutdown flow, disconnect handling, and client recovery

### In Memory Cache | Python, Gevent, TCP Sockets, Redis Serialization Protocol

[GitHub](#)

- Achieved 7.1K ops/sec with success under load, sustaining 6K+ ops/sec with up to 20 concurrent workers
- Optimized request path for 0.26ms avg GET latency and 0.34ms avg SET latency in single-threaded tests
- Implemented RESP over TCP sockets with async networking, supporting GET, SET, TTL, and AOF persistence

## SKILLS

**Languages:** Python, C++, Java, SQL, Bash

**Technologies:** Kafka, Redis, PostgreSQL, MongoDB, FastAPI, Docker, Linux, Git, Grafana, Humio

**Interests:** Distributed systems, Networking, Infrastructure