Zeus James Baltazar

Software Engineer

https://linkedin.com/in/zeus0390 | work@zeusjames.com | +639323831435 | San Jose del Monte, Bulacan, Philippines

Summary

Software Engineer specializing in C++ and backend systems. Skilled in modern C++, Python, multithreading, debugging, and performance-oriented development. Seeking backend, systems, and distributed software engineering roles focused on building and improving software products.

Technical Skills

C, C++, Python, SQL, Git, Unix/Linux, SSH, CMake, Docker, gdb, gprof, ASan, TSan, multithreading, vim, tmux, Azure

Professional Experience

Software Engineer - *Amdocs Philippines Inc*
Taguig, Metro Manila, Philippines | August 2024 – June 2026 | Hybrid

- Maintained and enhanced software and automation solutions, including modifications to C++ codebases, supporting complex workflow and data-processing requirements across Oracle/PostgreSQL databases, APIs, and Linux-based environments.
- Performed root-cause analysis of software issues by investigating existing codebases, logs, system behavior, and dependencies, and implemented targeted fixes and improvements.
- Collaborated within Scrum development teams to implement and deliver software changes and features while tracking development tasks using Jira and Confluence.
- Utilized Python, Java, Bash, KornShell, Groovy, and SQL to support system development and integration.
- Implemented and optimized existing system functionality while adhering to organizational best practices and engineering standards.

Educational Attainment

Bachelor of Science in Computer Engineering
Technological Institute of the Philippines - Quezon City | May 2018 – Sep 2023

Portfolio Projects

Peer-to-Peer File Sharing Network System

- Developing a peer-to-peer file sharing system in C++ for distributed network communication.
- Built the core networking layer with dual-stack IPv4/IPv6 support and event-driven I/O.
- Applying modern C++ techniques, including STL, RAII, smart pointers, mutexes, lock-based synchronization, multithreading, and design patterns to improve performance, resource efficiency, and maintainability.
- Currently implementing peer discovery and file transfer capabilities.

Adaptive Time-Window Chunking with Threshold-Bound Querying

- Developed a Python-based automation tool to efficiently retrieve large datasets from the company's Elasticsearch API while adhering to the 10,000-record query limit.
- Implemented a binary-search–style algorithm with dynamic sliding time windows to automatically calculate optimal date and time ranges for data extraction, eliminating manual adjustments and improving retrieval efficiency by 98%.

Professional Development

Beginning C++ Programming - From Beginner to Beyond, Udemy