Kunal Chand

New York, USA | 716-292-5504 | kchand@buffalo.edu | LinkedIn | GitHub | Portfolio

Skills

Languages: Java, C, C++, Python, HTML, CSS, SQL, JavaScript, PHP, Go, Shell.
Tools and Frameworks: Git, Jira, Maven, Linux, AWS Cloud, Spring Boot, Spring MVC, Selenium.
Functional Skills: Object Oriented Programming, Pair Programming, Feature Driven Development, SDLC, Agile Model.
Miscellaneous: Data Structures, Algorithms, RDBMS, JDBC, TCP/IP, JSP Servlets, J2EE, MPI, OpenMP.

Education

University at Buffalo, The State University of New York — **Buffalo, New York, USA**
Master of Science in Computer Science and Engineering — *Aug 2022 – Dec 2023*

Kalinga Institute of Industrial Technology (KIIT University) — **Bhubaneshwar, Odisha, India**
Bachelor of Technology in Computer Science and Engineering — *Jul 2016 – May 2020*

Experience

Volunteer Research Assistant — **Feb 2024 – Present**
University at Buffalo (SUNY) — *Buffalo, New York, USA*

- Executed molecular dynamics simulations of protein systems to analyze stability, including porting existing molecular codes to supercomputers, optimizing performance, and developing parallel algorithms for computational analysis.

Student Assistant — **Aug 2023 – Dec 2023**
University at Buffalo (SUNY) — *Buffalo, New York, USA*

- Facilitated over **70 students** in learning Distributed Systems concepts by providing hands-on assistance in comprehending theoretical concepts and **troubleshooting practical code** implementations in Go language.
- Utilized plagiarism detection tools and prepared detailed reports to identify academic integrity violations.

Associate Software Engineer — **Aug 2020 – Aug 2022**
NRI (Nomura Research Institute) FinTech — *Kolkata, West Bengal, India*

- Enhanced user experience by contributing to developing interactive fintech **portal** that provides a visual summary of monthly financial transactions using **Java Spring Framework** & eclipse debugger to **identify performance bottlenecks**.
- Assured data integrity by aiding in the batch/service development for reconciliation of the front-office financial data with the back-office financial data by using techniques like **code refactoring** and Test Driven Development (**TDD**).
- Improved code quality and adherence to project requirements by **conducting over 40 code reviews** before code merge.
- Leveraged SQL skills to enhance and contribute to the project's database design and functionality by crafting Data Definition Language (**DDL**) and Data Manipulation Language (**DML**) scripts ensuring optimal database performance.
- Ensured robustness and reliability of the software by executing approximately **6000 unit test** cases and validated the interaction between different software modules by running around **300 integration test** cases.
- Collaborated with team members through version control such as Git to organize modifications and assign tasks.

Projects

NRIFT Selenium | *Java, Selenium, Eclipse* — GitHub Link

- Built a browser automated solution using Java to **streamline the application process** on the in-house attendance app.
- Implemented selenium to create an instance of chrome in order to interact with the correct elements of the web page.
- Designed a **one-click** attendance application tool which automates the **end-to-end workflow**, reducing manual efforts.
- Optimized efficiency by mechanizing attendance application, resulting in **75 percentage time savings** per submission.

Parallel Systems | *C/C++, MPI, OpenMP, Slurm, Shell* — GitHub Link | Demo Link

- Implemented a parallel processing algorithm in C/C++ having distributed and shared memory to count the number of subset sums equaling a target value, achieving a **5x** speedup over sequential execution for different large inputs.
- Incorporated a **load balancing** feature that dynamically divides workload between MPI processes and OpenMP threads, by adjusting column-wise decomposition of dynamic programming table at runtime, leading to a **400% efficiency gain**.
- Streamlined testing via Shell script to submit Slurm job on cluster for each test case, **reducing manual efforts by 90%**.