Rafay Usmani

rafayusmani13@gmail.com | +1(331)-643-8814
https://www.linkedin.com/in/rafay-usmani/| https://github.com/rafayusmani13

PROFESSIONAL SUMMARY

Computer Engineering student interested in enhancing my skills by getting experience working at internships related to Software Engineering, Full Stack, Back-end, Front-end, Embedded Systems, and Cloud.

SKILLS

- Languages/Frameworks: C++, C, Java, Python, HTML/CSS, JavaScript, Assembly, Dart, Node, Django
- Tools: Flutter, Git, GitHub, Tiva C, Jest, REST API's
- ECE Concepts: Systems Architecture, Embedded Systems, Circuit Analysis, Microcontrollers
- Engineering Practices: Test-Driven Development, Code Reviews, Teamwork

EDUCATION

University of Illinois at Chicago, College of Engineering Chicago, IL
Bachelor of Science – Computer Engineering May 2022

- Engineering Coursework: Data Structures, Systems Architecture, Object Oriented Programming, Circuit Analysis, Embedded Systems, Logic Design, Discrete Math

WORK EXPERIENCE

Optimity San Francisco, CA
Software QA Engineering Intern October 2020 - Present

- Utilized JavaScript and Jest to write and automate test scripts for Optimity's REST API.
- Performed daily manual and AI testing on the Optimity app using Kobiton, as well as focused feature testing before new releases.

Genuine Commerce Ann Arbor, MI
Full Stack Software Engineering Intern May 2020 – October 2020

- Worked on Magnify, a searching, archiving, and reporting tool that delivers visibility for critical business documents such as bank wires and purchase orders.
- Utilized HTML, CSS, Django, and JavaScript to create the documentation website for Magnify. The website dynamically generates content and updates as changes are made.
- Used Django to integrate client's transaction databases to the Magnify app, which then consolidates data into a user-friendly interface.

PROJECTS

Custom CPU Design October 2020

- Created a MIPS-inspired Assembly subset, featuring special instructions such as register width calculation and program halting.
- Programmed an IDE for aforementioned Assembly in Python, which reads and compiles machine code input and returns various information, such as register, data memory contents, and cache hit/miss data.
- Designed a custom CPU, including a custom Instruction Memory, Data Memory, and instruction execution.

Pathfinding Algorithm Visualizer December 2020

- Implemented various pathfinding algorithms using Python, notably A* and Dijkstra.
- Created an interactive GUI using Pygames, allowing users to choose start/end locations on the grid, as well as create custom path barriers.
- Ported the tool into a webapp, utilizing Django for the front-end and Flask for the back-end, to visualize and compare the effectiveness of different pathfinding algorithms.

Spam Filter Program January 2020

- Implemented a spam filtration system for emails, using C++ on the Linux subsystem and Valgrind to ensure proper memory usage.
- Ensured an efficient O(N * log N) runtime by utilizing vectors and structs as primary data structures.