HARSHIT JAIN

+1 (582) 203-9755 ◇ harshitj.cs@gmail.com ◇ linkedin.com/in/harshitjain17 ◇ github.com/harshitjain17

EDUCATION

The Pennsylvania State University, University Park PA Expected Graduation: Dec 2024
Bachelor of Science in Computer Science
GPA: 3.7/4.0 | Dean's List (5/5) | Recipient of The President Walker Award | AlgoPSU Captain @ ACM | Resident Assistant
Relevant Coursework: Data Structures & Algorithms, Artificial Intelligence, Supervised Machine Learning, Advanced Learning Algorithms (Deep Learning), Systems Programming, Operating Systems Design, Theory of Computation, Programming and Computation II: Data Structures, CodePath: (Intermediate + Advanced) Software Engineering, Database Management Systems

TECHNICAL SKILLS

Programming Languages: Python, C/C++, JavaScript, Java, HTML/CSS, MATLAB, Verilog, Assembly (64/32-bit x86)
Frameworks & Tools: AWS, TensorFlow/Keras, Scikit-Learn, Numpy, Node.js, React.js, Next.js, RESTful APIs, LaTeX, Git
Softwares: MS SQL Server, MySQL Database System, Linux/UNIX, SonarQube, Postman, Bitbucket, JIRA

WORK EXPERIENCE

Machine Learning Engineer Intern Jan 2024 - Present
2D Crystal Consortium - Materials Innovation Platform (2DCC-MIP, MRI), Penn State University University Park, PA

- Working on the **ML-centric development** of "MaterialsTube", a video aggregator platform for data-driven material discovery. Leveraging **AWS (backend)** and integrating **ML models** for enhanced video metadata and content analysis
- Utilized: Python, AWS, Next.js, TensorFlow, Scikit-Learn, Deep Learning Models

Software Engineer Co-op May 2023 - Dec 2023
VIAVI Solutions Inc. Germantown, MD

- Collaborated with the 6-person R&D team to design and implement a **Python-based automated test suite** on Linux systems for the PNT instruments, ensuring comprehensive test coverage and compliance with the SCPI protocol
- Debugged PNT unit's source code in **C/C++**, resulting in a **55% reduction** in bugs and a **30% increase** in code coverage
- Performed **35+ short-term and long-term tests** on core devices using SCPI commands to uphold release-level quality
- Utilized: Python, C/C++, SCPI Protocol, Bitbucket, Confluence, SonarQube, Git, Agile, JIRA

Software Engineer Intern - Research Associate May 2022 - May 2023
2D Crystal Consortium - Materials Innovation Platform (2DCC-MIP, MRI), Penn State University University Park, PA

- Implemented **front-end architecture using React.js** to design **50+ user-facing features** in 20+ REACT components with 100% accuracy (tested using JEST), built reusable components, and front-end libraries for continuous development
- Integrated **MS SQL Server** relational database which currently deals with **500+ instruments' data** in 18+ tables
- Developed and tested **Python scripts** automating the process to retrieve data from various sources, manipulate and analyze the data, filter out irrelevant data, look up similar data in the server, import it into the server, and handle errors gracefully
- Developed and **automated a Python library** for the Raman Fitting model, to perform deconvolution on Raman spectra, and enable interactive preprocessing, effective fitting, and export of data files, **reducing analysis time by 40%** (tested)
- Utilized: Python, JavaScript, React.js, Node.js, MS SQL Server, RESTful APIs, Git, HTML/CSS, JIRA

PROJECTS

Dynamic Memory Allocator [C/C++] (Code available upon request; to avoid plagiarism) Mar 2023 - May 2023

- Designed **custom malloc, free, realloc**; segregated free lists and footer optimization to improved memory management
- Achieved a **utilization score of 69%** and benchmark **throughput at 100%** across diverse computing environments

HiLite: AI AutoHighlighter [Python, Flask, React.js] (Try it here) Mar 2023 - May 2023

- Designed an **AI system** that summarizes text using **Long Short-Term Memory (LSTM)** networks
- Created **LSTM-based Encoder and Decoder** to create a robust text summarization solution

mdadm Linear Device [C/C++, Linux] (Try it here) Feb 2023 - May 2023

- Configured **16 disks of size 64 KB as a 1MB** linear device, providing users with a unified address space for data access
- Implemented **mount/unmount operations** to the linear device, mitigating potential data loss and system crashes
- Designed the **read/write functions** to set up in the linear device, providing users with comprehensive data access capabilities
- Engineered **data caching solution** to enhance system latency **reduced I/O wait time by 60%**

Library Management System [Python] (Try it here) Nov 2021 - Dec 2021

- Built a **Python**-based library management system optimizing advanced tasks in a library setting; passed tests on **500+ logs**
- Deployed **10+ advanced features** and functions, including real-time student eligibility checks for book borrowing according to their historic data, calculated pending fines at the end of the log and on a specific day within the log, and more