Parth Sinha

Somerset, NJ • (732) 895-1282 • parthsinha03@gmail.com
Linkedin • Github • Website

EDUCATION

Rutgers University, New Brunswick, NJ **Expected Graduation May'25**
B.S in Computer Science (School of Arts and Sciences)
Accolades: Chancellor's Scholarship, Scarlet Scholarship
Relevant Courses:

- Algorithms
- Data Structures
- Machine Learning
- Deep Learning
- Linear Optimization
- Database Management

TECHNICAL SKILLS

- **Programming & Scripting** - Python, React, Java, Javascript, C, C++, HTML, SQL, Tailwind, SvelteKit
- **Tools** - IntelliJ, Git, VS Code, PostgreSQL/MySQL/MongoDB, AutoCAD, TouchDesigner
- **ML/DS** - Numpy, Pandas, Pytorch, NLTK

WORK EXPERIENCE

Web Developer Intern at Resilience Inc. **October 2024 - February 2025**

- Managing and updating mylearningtools.org and myselena.org, resolving errors, updating plugins, and uploading content to promote social emotional curriculum.
- Collaboration with the IT department to develop projects, utilizing WordPress to enhance website functionality.
- Analyzing technical requirements to determine user needs and resolved conflicts within teams to prioritize content criteria.
- Designing and implementing custom features using HTML and CSS to create clean website layouts.

Amazon Warehouse Operations Specialist **June 2021 - September 2021**

- Worked with Amazon's warehouse management system to ensure accurate tracking of inventory and timely delivery of products
- Coordinated with cross-functional teams to optimize warehouse operations and improve efficiency

Mechatronics Technician (Student Program) **September 2017 - June 2021**

- Worked in the Mechatronics, Engineering and Advanced Manufacturing (MEAM) program in Raritan Valley Community College and trained with the skills and knowledge to create tools, parts, and objects by operating milling and drilling machines, grinders, and lathes in a machine shop
- Earned OSHA 10-Hour Safety Certification and NIMS Certifications.
- Applied advanced skills in robotics by building a fully functional robot using VEX components and programming it with EasyC.
- Designed using Autodesk Fusion 360 for prototyping and assembled a robot car equipped with ultrasound sensors, programmed to navigate autonomously and controlled via a remote controller.

PROJECTS

MERN RealChat App **March 2025**

- Developed a full-stack real-time chat messaging app using the MERN stack (MongoDB, Express, React, Node.js) for instant messaging and user interaction with friends.
    - Built a responsive and dynamic frontend with React and Tailwind CSS, incorporating React Router for multi-page navigation and custom hooks for state management.
    - Implemented JWT-based user authentication and profile management with Cloudinary for profile picture uploads and storage.
    - Integrated PostCSS and Vite to optimize CSS processing and development workflow.
    - Node.js and Express.js to create a backend architecture, implementing efficient server-side routing, middleware for authentication, and RESTful API endpoints along with Socket.io for real-time messaging
    - Optimized client–server communication by configuring CORS and developing a responsive React.js frontend interface, resulting in a seamless user experience across devices.

Web Portfolio **February 2025**

- Developed a fully responsive personal portfolio website using React.js and Tailwind CSS, showcasing projects, skills, and contact information.
    - Implemented dynamic UI components with reusable Tailwind classes, ensuring a modern and accessible design.
    - Integrated Framer Motion for smooth animations and interactive elements to enhance user engagement.

Bomb Lab **November 2023**

- Reverse-engineering and debugging project to defuse a binary bomb by analyzing its assembly code and providing correct inputs for each phase
    - Utilized GDB to step through lines of assembly code, set breakpoints, and examine specific registers to understand the logic of the binary bomb
    - Employed reverse engineering using C language to understand high-level logic that corresponds to the low-level assembly instructions such as control structures and direct memory access for data manipulation
    - Debugged C code on GNU assembler to see which functions are called and which local variables and arguments are being called from the stack

Game of Life **October 2022**

- Java implementation of John Conway's "Game of Life" that progresses the state of the game's board through its many iterations
    - Utilized disjoint-set data structure to efficiently determine and manage connected components on the game board.
    - Employed a quick union-find algorithm to optimize the connectivity of cells and update their states based on the rules of the game.
    - 2D arrays, lists, and sets are used to represent the grid of cells, to keep track of the cells that need to be updated in each generation, and to accurately lookup and store live cells, respectively.