

Srujan Ponnappalli

Raleigh, NC | sponnap2@ncsu.edu | linkedin.com/in/srujanponnapalli | github.com/srujanponnapalli | srujanponnapalli.com

Computer Science, Economics, and Mathematics triple major at NC State who ships tested software: a Python desktop tool built for release to research institutions managing petabytes of data, a React and TypeScript web app that checks model output against retrieved sources, and systems code in C and Java.

Education

North Carolina State University

B.S. Computer Science, B.S. Economics (Data Analytics Concentration), B.S. Mathematics

Raleigh, NC

Expected May 2028

South Piedmont Community College

Associate in Arts; Certificate in Business Administration

Monroe, NC

May 2024

Experience

Software Engineer Intern

June 2026 – August 2026

Renaissance Computing Institute (UNC-Chapel Hill)

Chapel Hill, NC

- Built and shipped a cross-platform desktop application (Python, PySide6, Watchdog, REST API) that watches a researcher's local folders and uploads new or changed files automatically, replacing hand-run command-line transfers, for release to universities, national laboratories, and companies managing petabytes of data.
- Wrote its pytest unit suite from scratch, testing ingest behavior against the live research-data platform the tool uploads into, and diagnosed transfer failures that users reported against the running system.
- Delivered it on an agile team: daily standups, code review given and received, fast pull-request turnaround in Git, and a closing presentation of the design decisions to institute leadership and the member organizations that run the platform.

Undergraduate Researcher

January 2026 – Present

North Carolina State University (AIMS Laboratory)

Raleigh, NC

- Building physics-informed graph neural networks in Python that predict how long PFAS “forever chemicals” persist in the environment, and a feature pipeline (RDKit) that turns 14,000 chemical structures into the numeric fingerprints those models train on.

Machine Learning Research Intern

May 2024 – September 2024

University of North Carolina at Charlotte

Charlotte, NC

- Deployed a YOLOv7 detection model onto the embedded computer of a mobile robot, shrinking it until object and face detection ran on the robot itself instead of in the cloud, and fed its output into the robot's decision loop alongside the robotics and systems teams.

Projects

Paper Trails: Multi-Model Media Verification App | TypeScript, React, Next.js, Amazon Bedrock

February 2026

- Built the full stack of a web app with a team of three at Hack NC State 2026: a user submits a photo, news link, or written claim and gets back a verdict, a timeline of where it has appeared online, and the sources behind both (React, three API routes).
- Integrated six external services called in parallel behind per-call timeouts: reverse-image and news search, professional fact-check ratings, and three language models across Amazon, Google, and OpenAI, degrading to a narrower evidence set when a provider fails.
- Held those models to the retrieved search results so every date and link in a verdict traces back to a real source, ranked the fact-check ratings above model output in a deterministic override ladder, and escalated only uncertain verdicts to the second and third models.

Surveillance Manager: Object-Oriented Data Structure Library | Java, JUnit; NC State CSC 316

November 2025

- Implemented a Java tool that finds every contact within N degrees of separation from a given phone number in a set of call records, writing both a breadth-first traversal and six map structures underneath it (unordered linked list, search table, skip list, splay tree, red-black tree, linear-probing hash). Timed all six from 64 to roughly one million records: the list-backed map blew a 30-minute timeout on the largest input and the linear-probing hash map finished it in under 15 minutes.

Snark: Block-Compressing File Archiver | C, POSIX I/O, Linux; NC State CSC 230

November 2025

- Wrote a command-line archiver in roughly 1,300 lines of C: a custom binary format with add, remove, extract, and list operations, byte-pair-encoding compression over 16 KB blocks, manual memory and file I/O management, and unit plus end-to-end tests.

Wealth & Wait Times: Raleigh 311 Analysis | Python, pandas, GeoPandas, scikit-learn

July 2026

- Extended a two-person course project into an independent study of whether wealthier neighborhoods get faster city services, building a pipeline that joins 11,700 Raleigh service requests to Census demographics. Found 1,601 still-open requests, the longest-waiting cases, silently dropped as missing data; counting them turned a null result into a measurable one, now being presented to the City of Raleigh.

Skills

Languages: Python, Java, TypeScript, JavaScript, C, SQL

Frameworks & Libraries: React, Next.js, Tailwind CSS, PySide6 (Qt), pytest, JUnit, PyTorch, scikit-learn, pandas, GeoPandas

Tools & Platforms: Git, GitHub, Linux, Docker, AWS (Bedrock), Vercel, AI-assisted development (Claude Code, Cursor, Codex)

Other: Retrieval-augmented generation (RAG), multi-model pipelines with fallback, REST API design

Honors & Leadership: Park Scholarship (competitive full-merit award for scholarship, leadership, and service), Dean's List; Internal Vice President, Packonomics (NC State Economics Club); Head of Finance, Krispy Kreme Challenge Board, overseeing the budget for a student-run charity race that raised \$160,000 for UNC Children's Hospital; Recruitment Committee, Service Raleigh (400+ volunteers)