Sachin Iyer

sachin@sachiniyer.com

https://sachiniyer.com
https://github.com/sachiniyer

Education

New York University   Sep 2020 — May 2023
Bachelors of Science in Computer Science   *Brooklyn, NY*

Basis Independent Silicon Valley   Sept 2017 — May 2020
High School Diploma   *San Jose, CA*

Experience

AWS Bedrock   Software Development Engineer   *Seattle, WA   12/2023 — Present*

Bedrock: Creating Generative AI infrastructure for AWS Bedrock. Part of the core team and launched/maintained key features like batch inference, model distillation, and provisioned throughput with AWS-Wide and ReInvent visibility.

Amazon Last Mile   Software Development Engineering Intern   *Seattle, WA   05/2022 — 08/2022*

Data Aggregation Service: Created a full stack service to visualize last mile delivery data. Created a Typescript React frontend with Polaris styling that calls an AWS backend implemented with Java Lambda Functions, API Gateway, S3, and Internal Amazon Services. Used S3 Select and Spark to filter through about 1TB per query.

NYU High Speed Research Network (HSRN)   Academic Researcher   *Brooklyn, NY   02/2021 — 05/2023*

Parallel File System: Deployed an NSF funded 6PB storage PFS (SeaweedFS) for usage internally and externally to the HSRN. Automated Deployment with Ansible Playbooks and Rust CLI. Benchmarked with Bonie++ and IOR.

Audio Conferencing: Created an audio service (Portaudio) in C++ that interfaces with internal broker service.

CI/CD: Developed documentation, linting, testing, deployment (DinD with Kaniko) pipelines for the project in Gitlab

Mentorship: Leader of the student research arm. Managed and onboarded over 110 students over 4 semesters

Hewlett Packard Enterprise (Aruba Networks)   Cloud Intern   *Santa Clara, CA   06/2019 — 08/2019*

Estimating Bandwidth: Estimated bandwidth using Auto ARIMA/Prophet and other time series algorithms.

Sparkup   Software Development Engineering Intern   *Brooklyn, NY   09/2022 — 12/2022*

UX Development: Implemented a new feature in React Native App to link names and phone numbers in transactions

Dark Forest   Graphics Intern   *San Jose, CA   12/2021 — 02/2022*

Shader Development: Created a typescript plugin that allows for custom WebGL shaders in the Dark Forest game.

Projects

Ansible Batch Runner: Used Rust and Clap to create a cli for batch running and managing Ansible Playbooks

Personal Blog: Put up a custom themed hugo blog git synced from through custom docker container builds/deploys site

Book Recommendation Engine: Wrote a recommendation engine to group books based on wikipedia page similarity.

Circular Buffer: Wrote a header only circular buffer library in an STL style (e.g. templating, custom iterators).

K3S Cluster: Created a portable resilient fault-tolerant k3s cluster that networks through a wireguard mesh (headscale)

College Rank: Extensible meta-ranking from conference proceedings/best papers, placement rank, paper age, interests

Control Display: Created a mass controllable (over 50 simultaneous people) LED matrix. Host built with arduino metro and platformio, USB Serial Buses for communication, and React, Docker, and Material UI for site

Corelink Audio Conferencing: Created an audio service (Portaudio) in C++ that interfaces with Corelink.

CTF: Built CTF with Docker Compose, Dnsmasq, Postgres, Node MQTT server, Rust/Tide HTTP server and XtermJS

Dark Forest Shader Development: Created a typescript plugin that allows for custom WebGL shaders in the Dark Forest game.

Delivery Service: Created a web crawling service to notify available grocery delivery slots for Whole Foods, Costco, or Safeway. Used AWS SNS, ECS, and Lambda functions. Created headless Chrome docker containers running Selenium.

Diff Challenge: Became the 3rd finisher of the diff challenge by ggerganov (ggml, llama.cpp, whisper.cpp, kbd-audio)

Electronic Trombone: Made a trombone midi controller using Arduino with capacitive sensing and pid controllers.

Foot Pedal: Built a portable guitar pedal using LiPo batteries, teensy 4.0 (and audio shield), LCD, and custom effects

Git-sync Webhooks: Fork of git-sync with webhooks instead of polling. Supports IP whitelists, signatures, and secrets

IGJ Title Analysis: Analyzed the titles of the Indian Geographical Journal for importance using the tf-idf algorithm.

Invoice Categorization: Automatically sorts invoices into categories with Bedrock Batch, Flask, React, Tailwind. Wrote a preprocessing step with a common crawl adapter to supplement data about companies. Also dogfooded my service.

IP Monitor: Used the QT Framework and built a KDE widget to monitor your public and private IP Addresses.

Link Checker: Created a Rust CLI tool with Clap to automate link verification from args, stdin, or files.

Parallel File System: Deployed an NSF funded 6PB storage PFS (SeaweedFS) for usage internally and externally to the NYU HSRN. Benchmarked with Bonie++ and IOR.

Mastodon Status: Created a Rust Lambda function with a custom Megalodon-rs to push outage status to Mastodon

mnist-wasm: A custom, resource efficient, rust/wasm nn with jit spawned web-workers, yew, axum, wasm-bindgen

Git OpenResty: Created a container that utilized Lua JIT with OpenResty to sync git repos as a git-sync alternative

Reactive Sign: Used AWS IOT, Lambda, and API Gateway to build a interactive LED sign through serverless infra

Resow: Created a better Craigslist free section with ReactJS, MUI, Open Layers, Express, MongoDB, S3, Mocha, Jest

Sembox: A drive with semantic searching over documents with blip, xsum, whisper, bert, mui/nextjs, by taking the cosine similarity of summaries/search terms. Supports images, text, video, audio through intelligent document type recognition

Apps Status: Built an API with Rust, Tokio Async, Axum, and Reqwest that proxies status for my self-hosted apps through a Rust Lambda function with a custom Megalodon-rs that pushes outage statuses to Mastodon every 5 min

Synesthesia Visualizer: Auditory Visual Synthesis Visualizer with librosa, yin, eks, flask, docker, blender, WebVR

Tweet Toxicity: Used DistilBERT, Pytorch, HuggingFace Transformers, Streamlit and AdamW to classify toxicity type

Personal Website: Self-deployed a BabylonJS website with kubernetes k3s and full CI/CD using Custom Git Sync

Wikipedia Editor: Used Spark, LDA, Cohere, NYU HPC/ SLURM for variance in editor topics with 6TB of dumps

Programming Skills

Langs: Typescript, Javascript, Node, C++, C, Rust, Java, Python, (e)Lisp, Bash/Zsh, Cuda, Kotlin, Perl, Lua, LaTeX

Tech: AWS, Linux, Emacs, QT, Docker, k8s/k3s Rancher, Nginx, Traefik, ReactJS, Tailscale, MongoDB, Postgres, . . .

Certifications

AWS Solutions Architect	*September 2021*
AWS Cloud Practitioner	*August 2021*
Stanford Machine Learning by Andrew Ng	*July 2020*
AWS Fundamentals Specialization	*June 2020*