

Amartya Singh

singhamartya99@gmail.com | +91 8902794545 | Kolkata, West Bengal

linkedin.com/in/amartyasingh1986 | github.com/AmartyaSingh07 | amartya-portfolio-ruby.vercel.app

Summary

B.Tech Computer Science graduate, 2026. I build backend services in Node.js and TypeScript. Wrote a Redis-backed API rate limiter that held around 290 requests per second under k6 load testing, and was the only developer on a five-person team's Next.js and Supabase product. Looking for a backend or full-stack engineering role and available to join immediately.

Skills

Languages: JavaScript, TypeScript, Python, Java, SQL

Backend: Node.js, Express, REST APIs, FastAPI, JWT authentication, Redis

Frontend: React.js, Next.js, Tailwind CSS, HTML/CSS

Databases: PostgreSQL, MongoDB, MySQL, Supabase (Auth, Row-Level Security)

Tools: Docker, Git, GitHub, Postman, Linux, Vercel, Render

Testing: Playwright, Vitest, k6

Coursework: Data Structures and Algorithms, Object-Oriented Programming, Operating Systems, DBMS, Computer Networks, System Design, Microservices

Projects

RateLockr | Distributed API Rate-Limiting Gateway

Node.js, Express, TypeScript, Redis, Docker, React

GitHub | Live demo

- Built an API gateway with two rate-limiting strategies, token bucket and sliding window log, in TypeScript and Redis. Under k6 load at 100 concurrent users it sustained about 290 requests per second and blocked every over-limit call across 3,000 requests.
- Moved counter updates into Redis Lua scripts so two gateway instances cannot double-count the same request. Replaced a wildcard SCAN with a fixed 30-bucket time series, which cut Redis round trips per request.
- Added a fail-open path so traffic keeps flowing when Redis is unreachable, and streamed live counters to a React and Recharts dashboard through a browser event bus.

Trashium | Waste Collection and Recycling Platform

Next.js, TypeScript, Supabase, Tailwind, Leaflet

GitHub | Live site | Sole developer on a 5-person team

- Built the full application in Next.js and TypeScript with three user roles, household, crew and admin, on Supabase Postgres. Access rules run as Row-Level Security policies in the database instead of only in the UI.
- Shipped the crew view with a live Leaflet map, zone-based pickup queues and geo-tagged proof photos served through short-lived signed URLs, plus an admin analytics dashboard in Recharts.
- Ran the team's pricing regression inside the booking request instead of hosting a separate inference service, backed by a Postgres fallback table so a quote never fails outright.
- Added English, Hindi and Bengali language support and made the app an installable PWA that works offline.
- Owned testing for the platform: Playwright end-to-end suites, Vitest unit tests, and alpha and beta rounds before each release.

Foggy Lane Detection | Computer Vision Pipeline

Python, OpenCV, NumPy, Streamlit

GitHub | Live demo

- Wrote a Python and OpenCV pipeline that removes fog from road images using dark channel prior, atmospheric light estimation and bilateral filtering, then detects lanes with Canny edge detection and the Hough transform.
- On low-visibility test clips the dehazed frames recovered lane markings that the raw frames missed entirely.
- Deployed it on Streamlit with automatic image downscaling, which roughly halved processing time per frame.

Education

Supreme Knowledge Foundation Group of Institutions, West Bengal

2022 – 2026

B.Tech in Computer Science and Engineering

Holy Home, Serampore

ISC 2021: 75.4%

Stepping Stone School, Rishra

ICSE 2019: 79%

Achievements and Certifications

- Smart India Hackathon 2024 and 2025: cleared the internal and institute rounds and reached the third of four national selection stages in both years.
- Unfold Hackathon 2022: national-level finalist.
- Solved 250+ problems on LeetCode and GeeksforGeeks.
- Copado Certified Fundamentals I and II (Salesforce DevOps) | Complete Web Development Bootcamp, Udemy | Introduction to Python, Coding Ninjas | Project Development Using Java, Udemy