

AAHAN SHARMA

☎ +91-7624814826 | ✉ aahan30.sharma@gmail.com | 🌐 aahansharma.dev | in notaahan | 📺 notblankz

EDUCATION

PES University

B.Tech in Computer Science — GPA: 8.76/10 (6 semesters)

Bengaluru, India

2023 – Present

- Awarded CNR-Rao Merit Scholarship for 4 consecutive semesters.

TECHNICAL SKILLS

Languages: Go, Python, C, JavaScript, SQL

Frameworks: Next.js, React, Node.js, Express

Tools & Platforms: Docker, Git, PostgreSQL, SQLite, MongoDB, Cloudflare (R2/Pages/Tunnels)

PROJECTS

forge | GoLang, Cobra

- Built a static site generator with a concurrent render pipeline using errgroup, and a live-reload dev server with debounced filesystem watching for sub-second rebuilds; generates and deploys aahansharma.dev.
- Implemented incremental builds via dependency tracking, rebuilding only pages affected by a change.

downpour | GoLang

- Built a concurrent multi-worker file downloader using HTTP range requests, sustaining 22.86 MB/s average across 32 workers on a 10 GB download.
- Designed a health-monitor goroutine that smooths per-worker throughput and restarts stragglers below $0.3\times$ the trimmed mean (15% each tail), with a 5s grace period for TCP slow-start and non-blocking restart signalling over a buffered channel.

shutterdev | GoLang, Next.js, Cloudflare R2, SQLite

- Built a self-hosted photo gallery with JWT-authenticated admin routes and a client-side worker queue that pipelines uploads to Cloudflare R2 object storage.
- Architected for zero-cost self-hosting: Cloudflare Tunnels expose the service from a home network without a static IP or open inbound ports, with SQLite and R2's free tier removing any managed-database or egress cost.

FlowerPQ | Python, Flower, ML-KEM, ML-DSA, gRPC

- Forked the Flower federated-learning framework to make client-server communication post-quantum secure, adding ML-KEM key encapsulation for direction-aware encryption of model tensors and ML-DSA-44 signatures on every gRPC message to prevent tampering and gradient-inversion attacks.
- Implemented the key-exchange handshake during node activation and per-message sign/verify across the transport layer, with backward-compatible fallbacks for unarmed nodes; validated with a pytest suite covering roundtrip encryption, tampered-payload rejection, and batch signing. Final-year project with a 4-person team.

zaptorrent | Python

- Implemented a BitTorrent client from the wire protocol up — tracker requests, peer handshakes, message framing, and piece assembly with SHA-1 verification.
- Coordinated concurrent peer connections with `asyncio` workers to download and reassemble pieces in parallel across multiple seeders.

BillSplit | React, Node.js, Express, MongoDB, Google OAuth

- Built a bill-splitting web app that parses uploaded receipt images to extract line items, prices, and quantities for per-person assignment.
- Generated shareable split summaries as downloadable images, with Google OAuth for session management.