

# Isaac Hayab

+234-707-622-5974 [github.com/naiih001](https://github.com/naiih001) [isaachayab0@gmail.com](mailto:isaachayab0@gmail.com)

## Education

### Federal University of Technology

Expected November 2030

Bachelor of Science in Computer Science (GPA: 4.88 / 5.0)

Minna, Niger

- **Relevant Coursework:** Data Structures and Algorithms(C), Prob. and Stat in CS (Python), Intro to CS II (C) , Linear Algebra w/ Computational Applications (Python)

## Experience

### Kat's & Co, Web3 Solutions

Jan 2024– Dec 2025

Backend Engineer

Remote, Canada

- Designed and maintained backend infrastructure handling 1,000+ monthly on-chain interactions with 99% transaction success rate.
- Built and deployed scalable Web3 services and smart contracts, eliminating 70% of manual processing overhead.
- Architected CI/CD pipelines and automated Linux deployments, cutting release cycles by 60% and production error rates by 40%.
- Collaborated cross-functionally with product and frontend teams to translate requirements into reliable, production-grade systems.

### Freelance

Feb 2022 – Nov 2023

Backend Engineer

Lagos, Nigeria

- Built Go and Node.js microservices and RESTful APIs serving 2,000+ users, improving response times by 35%.
- Optimized database queries and backend performance, reducing latency by 30% and error rates by 25%.
- Maintained PostgreSQL and SQLite-backed services with a focus on reliability and data integrity.
- Provided technical guidance and code review to collaborating developers on shared projects.

## Projects

### ProTask API | NestJS, PostgreSQL, Prisma, JWT, BullMQ, Redis, Docker

- Architected a multi-tenant team task management REST API with full RBAC — users, teams, projects, tasks, and comments with clean relational schema design across 6 normalized tables.
- Implemented JWT authentication, role-based guards (admin/member), and team invite flows with scoped access control per organization.
- Designed complex many-to-many relationships (Users ↔ Teams, Tasks → Projects → Users) with Prisma ORM — demonstrating production-level database modeling.
- Added activity logging, pagination, filtering, and rate limiting — features that separate hobby projects from production systems.

### PayFlow API | NestJS, PostgreSQL, Prisma, Paystack, BullMQ, Redis, Docker

- Building a SaaS payment and subscription backend with Paystack integration, webhook handling, and plan-based feature gating.

- Implementing idempotent webhook processing to safely handle duplicate payment events — a real-world reliability concern most junior devs skip.
- Designing subscription lifecycle management including upgrades, downgrades, trial periods, and background job-driven expiry via BullMQ.

## ***Technical Skills***

---

**Languages & Runtimes:** Go (Golang), Node.js, TypeScript, JavaScript, Python, C

**APIs & Architecture:** REST APIs, GraphQL (familiar), Microservices, WebSockets, System Design, RBAC, Auth & Session Management

**Databases:** PostgreSQL, MongoDB, MySQL, SQLite, Redis

**DevOps & Cloud:** Docker, Docker Compose, GitHub Actions, CI/CD, Linux, AWS (EC2, S3), Railway, Render, Fly.io

**Other:** Solidity, Web3 APIs, Smart Contracts, React.js, Git