

Prince Ayaata Awenanyame Kwame

Software Developer · Computer Science student · Accra, Ghana

princeayaata2@gmail.com +233 53 180 8553 github.com/Kali452345 linkedin.com/in/prince-ayaata-b2721a324

Computer Science student and software developer. I build native Windows tools in Rust and C++, an offline peer-to-peer Android library in Kotlin, and AI-powered web apps. I measure performance on real hardware, write down design decisions, and ship versioned releases.

EDUCATION

BTech Computer Science

Level 300 (third year), in progress

Accra Technical University, Accra, Ghana

EXPERIENCE

Intern, Soko Aerial

Current

Signal Training School, Burma Camp, Accra

PROJECTS

Flash · Kotlin library & app

v1.1.0 released · v2.0.0 in beta

Kotlin, Coroutines, Jetpack Compose, WebSockets, TLS 1.3, WebRTC · github.com/Soko-Aerial/Flash

- Built Flash, an offline peer-to-peer file transfer, messaging and voice/video calling app for Android that works over the local network with no server, internet connection or account.
- Designed it as 12 Kotlin library modules (discovery, network, transfer, security, persistence, messaging, calling and Compose UI) published through JitPack, including a transport-only path that avoids the encrypted database's native libraries.
- Implemented WebRTC calling that prioritises audio over video, with a quality governor that sheds video bitrate, then resolution, then the video stream when audio degrades.

Wallive · Desktop app

v1.0.0 released

Rust, Win32, DirectComposition, Direct3D 11, Media Foundation, Inno Setup · github.com/Kali452345/wallive

- Built a live video wallpaper for Windows 10/11 in Rust and Win32 that sends hardware-decoded frames straight to DirectComposition, measured at about 0.9% CPU for 1080p 30 fps playback.
- Designed an event-driven pause system (WinEvent hooks, power and session notifications, per-monitor coverage math) with no polling timers; memory falls from about 81 MB to 29 MB while paused.
- Confined all unsafe FFI to dedicated ffi.rs modules with documented SAFETY blocks, enforced by Clippy and tests in GitHub Actions CI.

CoreGaze · Desktop app

v1.0.0 released

C++, Win32, DirectX 11, Dear ImGui, DXGI, PDH · github.com/Kali452345/CoreGaze

- Built a Windows resource-monitor overlay in C++ with Win32, DirectX 11 and Dear ImGui that shows live CPU, RAM, GPU, disk and network telemetry in a click-through, always-on-top HUD.
- Implemented multi-adapter GPU monitoring from DXGI video-memory queries and PDH per-engine utilisation counters (3D, compute, copy, video decode and encode), matched to adapters by LUID.
- Kept the overlay cheap to run by rendering from cached metric snapshots instead of collecting metrics on every frame.

StudyMate AI · AI web app

Working app, deployable to Streamlit Community Cloud

Python, Streamlit, SQLite, PyMuPDF, python-docx, python-pptx · github.com/Kali452345/StudyMate

- Built an AI study assistant in Python and Streamlit that turns PDF, DOCX and PPTX lecture material into source-grounded quizzes labelled with the page, slide or section each question came from.
- Designed a provider layer across Ollama, Gemini, Groq and Hugging Face plus an offline mock provider, with automatic Gemini model fallback and local rate-limit tracking.
- Added AI grading of open-ended theory answers against model answers and key points, falling back to exact matching when the grader is unavailable.

DevPulse · AI web app

Working app

JavaScript, HTML & CSS, Netlify Functions, Netlify Blobs, Gemini, Groq · github.com/Kali452345/DevPulse

- Built a developer-news aggregator with a framework-free front end and 10 Netlify Functions pulling Hacker News, Dev.to, Lobsters, GitHub Trending and AI news into one feed.
- Wrote a serverless AI router that rotates across Gemini models and several API keys with per-key rate budgets, falls back to Groq, and caches responses for an hour.
- Added a scheduled function that picks the top new stories every six hours and generates articles stored in Netlify Blobs.

Quick Tools · Web app

Working app

Python, Flask, pypdf, pdf2docx, Pillow, Docker · github.com/Kali452345/Quick_Tools

- Built a Flask web app that bundles a QR code generator, PDF conversion, merging, compression and password protection, and image conversion and compression, packaged with Docker.

Project Starter · Developer tool

In daily use; not published yet

Node.js, JavaScript

- Built a zero-dependency Node.js CLI that scaffolds AI-agent operating manuals (AGENTS.md, CLAUDE.md), architecture decision records, and progress, error and handoff logs, with presets for web, Android and desktop projects.

SKILLS

Languages	Rust, C++, Python, JavaScript, TypeScript, Kotlin, HTML & CSS
Windows & native	Win32, DirectX 11, DirectComposition, Media Foundation, Dear ImGui, PDH / DXGI, Inno Setup
Web & backend	Astro, Node.js, Netlify Functions, Flask, Streamlit, SQLite, Docker
AI integration	Gemini, Groq, Ollama, Hugging Face, Multi-provider fallback, Prompt-to-JSON validation
Engineering practice	GitHub Actions CI, Cargo & Clippy, CMake, pytest, Architecture decision records, Benchmarking