Project name

Nku — Offline Medical AI for Pan-African Triage

Your team

W. Elorm Yevudza Jnr, MD/MS — Solo developer. Born and raised in Ghana. Incoming surgery resident, NewYork-Presbyterian Queens. MD/MS Columbia VP&S (2025); BA Neuroscience, Amherst College (2019). Maintains clinical connections with health professionals across Ghana for pilot coordination and field validation.

Problem statement

In Sub-Saharan Africa, fewer than 2.3 physicians serve every 10,000 people — far below the WHO's recommended 44.5 health workers [1,2]. Over 450 million people lack accessible primary care [3]. Community Health Workers (CHWs), the frontline of care, frequently lack reliable diagnostic tools due to equipment deficiencies and supply stock-outs [4] — yet nearly all carry Android smartphones [5].

Powerful clinical AI models exist, but require reliable cloud connectivity. In rural Sub-Saharan Africa, 25% of the population lacks mobile broadband entirely [6]. Cloud-based AI is impractical where it is needed most.

Target user: A CHW in a rural region with a $60+ TECNO or Infinix phone (3GB+ RAM) and no stable internet [7]. She needs immediate triage guidance — offline, on her existing device — to determine which patients require urgent referral. Transsion brands (TECNO, Infinix, itel) hold >50% of the African smartphone market [8].

Impact & Deployment logistics: Distributing a 2.3GB LLM to a CHW in a rural region is a primary logistical hurdle. We address this via a multi-tiered infrastructure strategy: 1. **Pilot Sideloading:** Supervisors provision phones centrally via MDM or side-load the model directly via MicroSD card, requiring zero village internet bandwidth. 2. **Play Asset Delivery (PAD):** The 50MB core app is installed via the Play Store. It automatically downloads the 2.3GB model as an `install-time` asset when the CHW intercepts 4G/LTE cellular connectivity in larger towns. 3. **Peer-to-Peer Viral Sharing:** African smartphone culture involves local peer-to-peer file transfer. Only one CHW per clinic needs to download the model via 4G; they can then use Android's native *Nearby Share* or *Xender* to beam the 2.3GB `.gguf` file to other CHWs' offline phones at ~30MB/s over Bluetooth or offline peer-to-peer protocols. 4. **Zero-Rated Data:** For scaled Ministry of Health rollout, the Google Play download URL is "zero-rated" through partnerships with major Mobile Network Operators (MNOs) (e.g., MTN, AirtelTigo), ensuring the massive download does not deduct from the CHW's personal cellular data balance.

For reviewers installing the APK directly: The app automatically detects if MedGemma is missing on first triage and downloads the 2.3GB Q4_K_M model from HuggingFace, validates its SHA-256 checksum, and proceeds — no manual setup required. Simply install the APK, connect to Wi-Fi, and run a triage.

Overall solution

Every line of clinical reasoning executes on the phone itself.

Nku (Ewe: “eye”) runs MedGemma entirely on $60+ Android smartphones. It is a proof-of-concept prototype; field validation with CHWs is the critical next step.

MedGemma 4B is irreplaceable in this system. It is the sole clinical reasoning engine, performing the interpretation that transforms structured sensor data and symptoms into triage assessments — a capability no smaller model possesses. Cloud inference fails completely in low-connectivity zones. Only MedGemma, quantized to Q4_K_M and deployed via llama.cpp JNI on ARM64, enables the offline + accurate combination Nku requires.

The Nku Cycle is a multi-stage orchestration pipeline where MedGemma serves as the clinical reasoning engine within a self-adapting workflow:

Stage	Component	Size	Function
1. Sense	Nku Sentinel (5 detectors)	0 MB	Camera + microphone → structured vital signs
2. Translate	Android ML Kit (On-Device)	~30MB/lang	Translates 59 supported local languages to English (offline). Unsupported indigenous languages are translated via cloud fallback; if no internet, they are not allowed.
3. Reason	MedGemma 4B (Q4_K_M)	2.3GB	Clinical reasoning (in English) on symptoms + sensor data
4. Translate	Android ML Kit	~30MB/lang	English → supported local language output (offline) or via cloud fallback
5. Speak	Android System TTS	0 MB	Spoken result in local language
Fallback	World Health Organization / Integrated Management of Childhood Illness (WHO/IMCI) rules	0 MB	Deterministic triage if MedGemma unavailable (e.g., insufficient available RAM)

Crucially, because optical sensors historically exhibit diagnostic bias against darker skin tones (classified as Types V and VI on the Fitzpatrick skin typing scale), every Nku camera modality in the “Sense” stage is engineered to be “Fitzpatrick-aware” to ensure equitable accuracy. Sensor confidence must exceed 75% for inclusion in MedGemma’s prompt; below-threshold readings trigger a localized warning prompting the CHW to re-capture in better conditions.

Adaptive Memory Management: While 3GB+ RAM is common in low-end Android smartphones, Android background apps often consume significant memory. Since third-party apps cannot programmatically suspend other processes, Nku implements an Adaptive Memory Management flow: before loading the 2.3GB MedGemma model, the engine queries the kernel (`ActivityManager.MemoryInfo`). The system specifically checks for ~1.2 GB of available RAM. This threshold acts as an empirical Safety buffer (Resident Set Size). Crucially, Nku relies on Android’s `mmap` implementation, which pages the 2.3GB model into the active virtual address space dynamically rather than loading the entire file into physical memory at once. By ensuring 1.2GB of breathing room, `mmap` can fluidly page the model through RAM during inference without "thrashing" (spending 100% CPU moving data between storage and RAM). If available RAM is below this threshold, Nku intercepts the flow with an alert: *"Insufficient RAM. Please close other apps or use standard guidelines."* Finally, if the OS OOM killer terminates the native Llama.cpp process during an unexpected load spike, Nku catches the `ENOMEM` exception and gracefully fails over to the deterministic WHO/IMCI rule-based triage.

Each stage operates independently. These built-in safety checks ensure the system never hard-crashes. All medical inference by MedGemma is 100% on-device and strictly reasons over English prompts for clinical safety. ML Kit provides on-device translation for 59 languages, ensuring that since CHWs are trained in their national official languages (e.g., English, French, Portuguese), a comprehensive 100% offline triage path is guaranteed. If a CHW selects an unsupported indigenous language, the app displays a UI connectivity alert: cloud translation is used as a fallback, and if there is no internet, the language is not allowed.

Before/after — why structured prompting matters: MedGemma was trained on clinical text, not smartphone sensor data. A prompt like *"the patient looks pale and her eyes are puffy"* yields generic advice. To address this, Nku's `ClinicalReasoner` fuses the CHW's input text with interpreted sensor data, feeding MedGemma a structured prompt containing quantified biomarkers and confidence metrics:

```
Conjunctival saturation: 0.08 (healthy 0.20, pallor threshold 0.10),
pallor index: 0.68, severity: MODERATE. EAR: 2.15 (normal 2.8, edema
threshold 2.2), edema index: 0.52. Patient pregnant, 32 weeks.
```

MedGemma's response to this structured input:

> `SEVERITY: HIGH | URGENCY: IMMEDIATE` — Identifies the classic preeclampsia triad (edema + headache + pregnancy >20 weeks), flags concurrent anemia, and recommends same-day facility referral with specific danger signs to communicate to the patient.

Previous studies have demonstrated that this structured prompting achieves a median 53% improvement over zero-shot baselines [9] — transforming MedGemma from a general medical QA model into a structured sensor data interpreter for CHW triage.

Technical details

Edge AI — Quantization & Memory: We achieve 69% model size reduction (8GB → 2.3GB) via Q4_K_M quantization while retaining 81% of MedQA accuracy (56% quantized vs. 69% unquantized baseline). The model runs on 3GB+ RAM devices via `mmap` — the OS pages model data on demand, so peak resident memory adapts to available RAM. We systematically benchmarked four quantization levels:

Quant	Size	MedQA	Primary Care	Verdict
Unquantized (Baseline)	8.0 GB	69.0%	-	Too large for RAM
Q4_K_M	**2.3 GB**	**56.4%**	**58.0%**	**Deployed**
IQ2_XS + imatrix	1.3 GB	43.8%	45.3%	Viable ultra-compact
Q2_K	1.6 GB	34.7%	33.9%	Worse than IQ2_XS
IQ1_M	1.1 GB	32.3%	32.4%	Near random

Each model evaluated single-shot on the full MedQA test set (1,273 questions) and the primary care subset (707 questions) — one attempt per question, no repeated runs or best-of-N selection.

Key finding: IQ2_XS with medical imatrix calibration outperforms the larger Q2_K by +9.1pp — domain-specific calibration matters more than raw bit budget. We created a 24-scenario African clinical triage calibration dataset across 14+ languages for imatrix generation.

Why not deploy the ultra-compact 1.3GB IQ2_XS? While saving 1.0GB of RAM and storage is tempting for rural networks, triage is high-stakes. The 12.6pp accuracy drop from Q4_K_M (56.4%) to IQ2_XS (43.8%) marks a critical threshold. As noted during our benchmarking tests of the quantized versions, the 1.3GB model loses its ability to reliably parse multi-morbidity (e.g., identifying concurrent pneumonia and severe malaria). Q4_K_M at 2.3GB is the smallest size we are comfortable using given the high-consequence triage environment. Furthermore, 3GB+ RAM is now widely available even on Android phones in the $60 bracket (e.g., itel A90, TECNO POP series), which represent the lower bound of devices sold in this market.

Nku Sentinel — Camera-Based Screening (0 MB additional weights): CHWs often lack equipment [4]. Nku extracts vital signs using only the phone camera via pure signal processing, then feeds structured biomarkers to MedGemma for clinical interpretation:

Screening	Method	Output	Fitzpatrick-aware
Heart rate	Green channel rPPG, 10s DFT [10,11,12]	±5 BPM	Adaptive thresholds
Anemia	Conjunctival HSV analysis [13,14]	Pallor score 0–1	Conjunctiva only
Jaundice	Scleral HSV analysis [15,16]	Jaundice score 0–1	Scleral tissue (unpigmented)
Preeclampsia	Facial geometry EAR [17,18]	Edema score 0–1	Geometry (color-independent)
TB/Respiratory	HeAR Event Detector pipeline [27]	Risk score 0–1 + health sound classification	Audio (skin-tone independent)

Why cough-based respiratory screening? Sub-Saharan Africa bears a disproportionate respiratory disease burden: 10.8M new TB cases globally in 2023, with only 44% of MDR-TB cases diagnosed and treated [1]. COPD prevalence in SSA is projected to rise 59% by 2050 due to biomass fuel exposure, prior TB, and weak diagnostic infrastructure [30]. Pneumonia remains the leading infectious cause of child death, claiming over 500,000 under-5 lives annually [31]. When a CHW suspects respiratory illness, they navigate to Nku's respiratory screen and record 2–5 seconds of the patient's cough or breathing. Nku then activates Google's HeAR (Health Acoustic Representations) Event Detector — a MobileNetV3-based classifier (1.1MB TFLite) that screens audio in ~50ms for 8 health sound events (cough, sneeze, snore, breathing, etc.) using a robust FP32 fallback architecture to ensure maximum device compatibility. Cough detection probability, event class distribution, and composite risk scores are passed to MedGemma for clinical reasoning.

Architectural Advantage of the Event Detector: Although traditional audio encoders output dense acoustic embeddings, MedGemma thrives on structured data. The 1.1MB TFLite

Event Detector rapidly classifies 8 specific health sound events (cough, snore, baby cough, breathe, sneeze, etc.) and outputs explicit confidence probabilities for each. Unlike traditional audio encoders that output dense embeddings, the Event Detector passes a structured summary (`Cough Probability: 0.82`, `Breathe Probability: 0.45`, `Risk Score: High`) directly into the `ClinicalReasoner` prompt. MedGemma is excellent at reasoning over these explicit probabilities alongside the patient's other symptoms. The Event Detector paired with MedGemma delivers meaningful clinical triage value: In settings where 44% of MDR-TB goes undiagnosed, even binary cough detection and breathing abnormality screening with clinical LLM reasoning provides a screening signal CHWs currently lack entirely.

When MedGemma is unavailable, the app displays a transparency banner identifying the triage as guideline-based (WHO/IMCI) with actionable recovery steps — all in the CHW's selected language.

Safety: 8-layer `PromptSanitizer` at every model boundary (zero-width stripping, homoglyph normalization, whitespace normalization, base64 detection, regex patterns, character allowlist, delimiter escaping, length capping). Auto-pause at 42°C. Always-on "Consult a healthcare professional" disclaimer. *Note on CVE-2025-69872 (`diskcache`): Since the HuggingFace Hub client pins `diskcache`, our optional inference API implements strict local directory restrictions and cache partitioning to mitigate unauthorized cache injection.*

46 Pan-African languages (14 clinically verified): ML Kit on-device for supported national languages (100% offline). Unsupported indigenous languages trigger a UI connectivity alert and use Cloud Translate fallback mechanics; all final reasoning occurs entirely on-device in English.

Prize Track: Main + Edge AI — Q4_K_M compression (8GB→2.3GB), mmap loading on $60+ phones (3GB+ RAM), llama.cpp JNI (NDK 29, ARM64 NEON), systematic 4-level quantization benchmark (IQ2_XS with medical imatrix calibration), 100% on-device inference with MedGemma bundled via Play Asset Delivery (2.3GB, install-time), and CHW-initiated respiratory screening via HeAR on-device: Event Detector (MobileNetV3-Small, 1.1MB TFLite with FP32 fallback) classifies 8 health sound events in ~50ms, with risk scores and event classes fed to MedGemma for TB/COPD/pneumonia triage.

Open source: Nku is fully open source under the Apache License 2.0. Source code, scripts, and calibration data on GitHub. Quantized model weights on HuggingFace (subject to Google Gemma Terms of Use).

See `kaggle_submission_appendices.md` for full references [1–32], language list (46), calibration scenarios, MedGemma reasoning examples, sensor pipeline details, and safety architecture.

Development tooling: *Google Antigravity (Gemini 3 Flash/Pro, Gemini 3.1 Pro, Claude Opus 4.5/4.6); OpenAI Codex IDE (GPT 5.3 Codex).*