

MENS MERCATUS GAZETTE

AI SYSTEMS · AGENTIC WORKFLOWS · PRODUCTION ENGINEERING

SERVICE DESCRIPTION

KATLEGO MOKOTO · AI ENGINEER / FULL-STACK .NET DEVELOPER

DEPENDABLE AI SYSTEMS, BUILT TO BE REVIEWED, DEPLOYED, AND TRUSTED

I design and build production-minded AI systems beyond simple chatbot prototypes: LLM applications, agentic workflows, retrieval-augmented generation, semantic search, AI automation, and full-stack products with observable behavior and human control.

I work across architecture, implementation, and deployment using Python, C#, ASP.NET Core, FastAPI, React, TypeScript, PostgreSQL, vector search, REST APIs, WebSockets, Docker, Linux, Ollama, Gemini, and OpenAI-compatible model interfaces. I integrate hosted foundation models or route requests to local and quantized models when privacy, cost, latency, resilience, or offline execution matters.

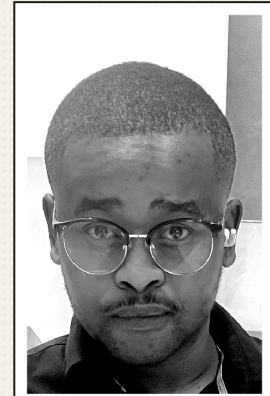


FIG. 1 — KATLEGO MOKOTO, AI SYSTEMS ARCHITECT. ON-SITE, HYBRID, OR REMOTE.

01 PRODUCTION DISCIPLINE

I build AI systems with provider abstraction, model and fallback routing, prompt and context management, structured outputs, retry and exponential backoff, rate limiting, health checks, logging, latency tracking, audit trails, and explicit failure states.

For agentic applications, I implement bounded tool use, state management, approval gates, replayable actions, and separation between model recommendations and irreversible operations.

02 RETRIEVAL & SEARCH

I develop retrieval systems using ingestion, chunking, metadata filters, embeddings, vector indexes, semantic similarity, content-hash caching, and model-aware retrieval.

One research desk uses Gemini embeddings when available and falls back to a local model when not. It re-embeds only changed documents and scores vectors only when they come from the same embedding model, preventing mixed-index searches from returning misleading rankings.

03 TOOL INTEGRATION

I integrated a 47-service remote tool plaza using SCP/MCP-style JSON-RPC, with handshake logic, protocol-version negotiation, SSE stream parsing, concurrent service sweeps, replayable tool-call records, and local fallback when the provider is unavailable.

The concurrent sweep achieved a measured 7.4× improvement over sequential execution.

04 SOUNDX BROWSER DAW

A 28-module custom Web Audio application with 18,000 lines of code. It includes scheduler-driven automation lanes, command-history undo/redo, strength-controlled grid quantization, interval-based chord generation, valid MIDI export, and an AI Producer that proposes changes without silently overwriting the session.

Its ffinpeg DSP layer performs EBU R128 loudness measurement, platform-target mastering, and spectral stem separation. The mastering flow was validated from -43.6 to -14.0 LUFS, with the output written and re-measured.

05 DELIVERABLES

- **Architecture** — system design and boundaries
- **Backend / frontend** — production code
- **Model integration** — hosted and local
- **RAG / semantic search** — full pipelines
- **APIs** — REST and WebSocket surfaces
- **Tool connectors** — MCP-style JSON-RPC
- **Vector stores** — indexes and filters
- **Local fallback** — quantized models
- **Deployment** — Docker, Linux, systemd
- **Documentation** — operational and handover

06 ENGINEERING STANDARD

I understand both ML application behavior and the production engineering required to make systems secure, observable, and maintainable.

Every deliverable ships with measurable behavior and a path to deployment — not a demo that falls apart under load.

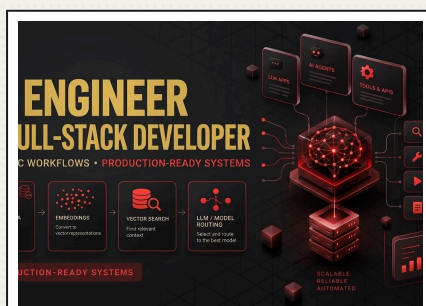


FIG. 2 — SERVICE ARCHITECTURE: DATA —

FROM MODEL, WORKFLOW, DATASET, OR PRODUCT IDEA TO A DEPENDABLE SYSTEM

If you have a model, workflow, dataset, API, or product idea that needs to become a dependable system, I can turn it into an implementation with measurable behavior and a path to deployment.

PYTHON

C# / ASP.NET CORE

FASTAPI

REACT / TYPESCRIPT

POSTGRESQL

VECTOR SEARCH

DOCKER / LINUX

OLLAMA

GEMINI

OPENAI-COMPATIBLE

HOW THE SYSTEMS ARE BUILT

AGENTIC WORKFLOWS
WITH A PERSON ON THE IRREVERSIBLE STEP

AI-assisted means a model returns a bounded output: a draft, a summary, a setting. Agentic means the system can pick an approved tool, keep state, and propose the next step. It does not mean the system sends mail, places orders, or hides a failure. The pipeline below is the same shape I apply to every build.

01 INTENT What the person asked for.	02 CONTEXT Only the data that task needs.	03 ROUTER Model, tool, or fallback.	04 TOOLS Approved APIs and files.	05 HUMAN GATE Stop before irreversible acts.	06 RECORD Status, latency, source.
--	---	---	---	--	--

```
KATLEGO@MERCATUS - ZSH

% tool-plaza sweep --concurrency 47 --json-rpc
handshake ..... ok · protocol v2024-11-05 negotiated
sse stream ..... parsed · 47/47 services reached
sequential ..... 12.6s
concurrent ..... 1.7s → 7.4× improvement
fallback ..... local model engaged when provider unavailable
% soundx master --measure ebu-r128
input ..... -43.6 LUFS
output ..... -14.0 LUFS · written and re-measured
% retrieval index --model-aware
re-embed ..... changed documents only
score guard ..... same-model vectors only · no mixed-index ranking
```

01 · CAREER PIPELINE

HUMAN APPROVAL REQUIRED

A daily career pipeline running a 20:00 SAST scrape, sentence ranker, and one-role-per-company rule. A note is dispatched only when the posting prints an employer address, that mailbox verifies, and the letter cites the role. Otherwise the send is held for human confirmation.

STACK: AGENTS · GMAIL · MACOS AUTOMATION · CV TAILORING

02 · MULTIMODAL ASSISTANT

LIVE, WITH A LABELLED FALLBACK

A .NET Core 8 service and React workspace calling Gemini from the server, keeping the API key off the browser. Provider-swappable chat completions, exponential backoff on 429/5xx, per-IP caps, systemd daemonization, and a labelled demo mode when upstream is degraded.

STACK: C# · ASP.NET CORE 8 · REACT 19 · TYPESCRIPT · GEMINI

03 · RESEARCH DESK

SESSION-SCOPED AND BOUNDED

A hosted terminal for charts, multi-market views, and a session-scoped paper bot. Compact footprint (~87 MB on disk), quotes refreshing near every 60 seconds, up to 24 concurrent sessions with a 2-hour idle TTL, and a public health JSON status endpoint.

STACK: PYTHON · HIGHCHARTS · YFINANCE · UBUNTU VPS

04 · IN-BROWSER STUDIO

CLIENT-SIDE EXECUTION

A guided four-step browser workstation: Load Demo, Play, Shape Master, and Generate. Audio processing executes entirely client-side with zero data egress and no required authentication. It listens to the audio buffer, estimates tempo from onset intervals, and requests user confirmation prior to application.

STACK: WEB AUDIO API · TYPESCRIPT · DSP · AI PRODUCER

ENGAGEMENT

Available for AI engineering, full-stack .NET, and product engineering teams. On-site, hybrid, or remote — Johannesburg, Pretoria, Cape Town, Durban, and remote teams. Typical engagements begin with an architecture pass, then move to implementation, deployment configuration, and documentation handover.