

MOHAMED SAMEH

AI Engineer | Computer & Systems Engineering, Ain Shams University

+20 101 457 7157 • mohamad.sameh.salem@gmail.com • linkedin.com/in/mohamed-sameh77 • github.com/mohamedsameh20

Professional Summary

AI Engineer specializing in multimodal AI systems, computer vision, and agentic workflows. Proven track record of architecting and deploying Machine Learning pipelines, Vision-Language Models (VLMs), and asynchronous real-time LLM agents.

Education

Faculty of Engineering, Ain Shams University
B.Sc. Computer and Systems Engineering – CGPA: **3.00**

Cairo, Egypt
2021 – Jun 2026

Projects & Experience

IntelliRoom — AI Generative Interior Design Platform *Graduation Project · Grade: A+* **2026 – Live**
ComfyUI, Stable Diffusion, Flux, LoRA, Vector DB, VLM, Python

- Designed and implemented **7+ generative AI workflows** covering inpainting, ControlNet layout locking, style transfer, LoRA fine-tuning, upscaling, and 3D asset generation.
- Built a multi-modal ComfyUI engine with a boolean mode-switch system for low-latency mode toggling, transitioning to Flux Flow-Matching models for spatial layout preservation.
- Automated object replacement by chaining **Florence-2** (detection), **SAM2** (segmentation), and **IP-Adapters** (texture injection) for zero-touch visual masking and re-texturing.
- Architected an e-commerce visual product retrieval pipeline using **RT-DETR**, **SigLIP2**, and **Qwen3-VL** embeddings, querying a **ChromaDB** vector store for real-time commercial matching.
- Integrated **SeedVR2** super-resolution for artifact-free upscaling alongside a **Hunyuan3D** pipeline to generate textured 3D .GLB furniture meshes from 2D renders.
- Deployed interactive platform for live generative design; authored complete technical documentation in $\LaTeX$.

Document AI Extraction App *1st Place · STP MACHATHON 7.0* **2026**
PyTorch, TATR, IBM TableFormerV2, PaddleOCR, ONNX Runtime, FastAPI, React

- Built an end-to-end table extraction pipeline fine-tuning **TATR** (detection/structure) and **PaddleOCR v5 mobile** (det+rec) with a custom geometric cell-assignment layer.
- Fine-tuned **IBM TableFormerV2** on new annotation criteria with dual-stream supervision (OTSL token sequences + cell-box regression, GloU+L1), reaching bbox **IoU 0.79**.
- Diagnosed a **5,000+ sample annotation bug** that tanked AP to 0.48; engineered a data repair tool that restored performance to **AP 0.90+ / AP50 0.95+**.
- Applied adversarial hard-negative training to eliminate false-positive table detections on non-table document layouts.
- Shipped a full-stack **FastAPI + React** editing studio with multi-format export (JSON/CSV/XLSX); containerized and deployed to Hugging Face Spaces.

M.A.C.E. — Multimodal Agent for Concurrent Execution *Research Project* **2026**
TypeScript, Bun, Cerebras API, libSQL, ONNX Runtime, Mineflayer

- Re-engineered NVIDIA's Voyager Minecraft architecture into a concurrent 3-agent system (Commander, Operator, Overwatch) on an in-process event bus, executing codegen at a measured median **0.52s latency** on Cerebras.
- Built a soft-real-time supervisory critic loop running concurrently every 3s, triggering emergency interrupts and aborting running skills mid-execution upon detecting environmental threats.
- Implemented an in-process skill library with a local 384-dim vector store (BGE-small ONNX + libSQL), achieving local code retrieval to reuse verified skills and minimize LLM calls.

MoaLLM — AI Learning Studio *Personal Project* **2026 – Ongoing**
Python, TypeScript, Node.js, MCP, Obsidian, libSQL, ONNX Runtime

- Built an **Obsidian learning workspace** connecting external AI agents through **6 MCP tools**, supporting interactive tutoring, handwritten-image submissions, and Markdown/LaTeX conversations.
- Implemented **local hybrid search** combining FTS5/BM25 and DiskANN vector retrieval over 384-dimensional ONNX embeddings with reciprocal-rank fusion; cached embeddings by content hash to skip redundant inference. Shipped a companion **Python** $\cup$ **CLI** for vault search, graph maintenance, source ingestion, and retrieval logging.

- Engineered an authenticated, **retry-safe messaging bridge** with idempotent requests, reply acknowledgements, and cancellation-aware long polling; covered transport, session isolation, rendering, and UI behavior with **91 automated tests**.

MalWhere — Agentic Malware Analysis Tool
Personal Project

Ongoing

FastAPI, LightGBM, SQLite, ReAct Agent, Llama.cpp, Python

- Built a local **ReAct agent** that dynamically inspects PE binaries, deciding analysis tools to invoke based on intermediate findings rather than a static pipeline.
- Integrated **EMBER LightGBM** classifier, **YARA** community rulesets, PE string parsers, and a grayscale binary visualizer as agent-callable tools.
- Connected live threat feeds (VirusTotal, MalwareBazaar) with **SQLite** hash-based caching to eliminate redundant API requests.

KotchAlna — Algorithmic Basra AI Engine & Telemetry Dashboard
Personal Project

2025 – 2026

Python, React 19, TypeScript, Pygame, Pytest, Bayesian Inference

- Formalized Egyptian Basra into a 719-line mathematical architecture, proving state as an invariant set partition (Γ) and resolving multi-card captures via subset-sum combinatorial algebra.
- Engineered a Bayesian opponent tracker with negative inference to deduce unheld cards from uncaptured discards, optimizing Jack preservation timing, point denial, and expected discard penalties.
- Built a 7◇ “Chameleon” Basra partition solver; verified engine with **271 automated tests** in Pytest and shipped an interactive **React 19 + TypeScript** dashboard with real-time AI thought telemetry.

Zakkir — Android Prayer Times & Azkar App
Open-Source · Live APK

2026

React Native, Expo, TypeScript, Android, Playwright

- Shipped an **Android prayer-times and Azkar app** (live APK, CI-built via GitHub Actions), with the same app also running on desktop and as a browser extension; **10 releases**.
- Offline-first: bundled **138 adhkar** across 8 categories, cached prayer times, and scheduled prayer reminders that work without connectivity.
- Arabic-first design with RTL, bundled Arabic fonts, and **100+ themes**; covered by **31 automated tests** in CI.

PhotoValidator — Image Validation Pipeline
Client Work · Ain Shams University IT Desk

2025

PaddleOCR, ConvNeXt, PyIQA, OpenCV, Python

- Delivered an automated photo validation pipeline for the Faculty IT desk at Ain Shams University, sorting student photo batches into valid, invalid, and manual review categories.
- Combined text detection, watermark classification (**ConvNeXt CNN**), border detection, and multi-metric quality assessment with an adaptive scoring system.

Honors, Awards & Achievements

1st Place Winner · STP MACHATHON 7.0

2026

Built an end-to-end Document AI table extraction pipeline using TATR TD/TSR, PaddleOCR v5 mobile, ONNX Runtime, and RapidOCR fallback.

AISA-ArabicFC Participant · ArabicNLP 2026

Ongoing

Competing in Arabic LLM function calling, function selection, argument extraction, and interpretable agent reasoning traces.

Data Analysis Certification · Top Leaderboard Ranking

2024

National Telecommunications Institute (NTI). Hands-on training in data cleaning, visualization, Power BI, and interactive analytics dashboards.

Technical Skills

- Languages:** Python, TypeScript, JavaScript, SQL, $\text{\LaTeX}$
- ML Frameworks:** PyTorch, HuggingFace, ONNX Runtime, LightGBM
- LLMs & Agents:** LangGraph, ReAct Agents, MCP, Hybrid Retrieval, Llama.cpp, Prompt Engineering
- Computer Vision:** OpenCV, PaddleOCR, SAM2, Florence-2, RT-DETR, SigLIP2, ConvNeXt
- Backend & Infrastructure:** FastAPI, Docker, REST APIs, Asynchronous Programming (`asyncio`), Node.js, React Native, Expo
- Databases:** SQLite, ChromaDB, Turso Vector DB
- Tools & Practices:** Git, Linux, Figma, Dataset Annotation & Labeling, E2E Testing (Playwright)