

ANAS MTAWEH

AI Product Builder | AI-Assisted Systems & Rapid Prototyping

Aley, Lebanon • +961 76 177 435 • anasanasmtaweh@gmail.com • linkedin.com/in/anas-mtaweh
• github.com/Anasmtaweh

PROFESSIONAL SUMMARY

CS graduate specializing in AI-driven product development. Expert in identifying technical gaps, architecting end-to-end solutions, and shipping working products rapidly using AI-assisted workflows. Combines system-level design thinking (LLMs, retrieval systems, full-stack architecture) with strong product intuition, outcome-driven validation, and fast iteration cycles. Leverages Claude, GPT, and AI tools as primary development medium.

TECHNICAL SKILLS

Product & Strategy: Problem Identification, System Architecture Design, Outcome Validation, User Feedback Integration, Rapid Prototyping, AI-Assisted Development

AI & LLM Applications: Prompt Engineering, LLM Integration, Retrieval-Augmented Generation (RAG), AI Agent Design, Context Optimization, Dynamic Prompt Injection

AI Tooling & Platforms: Claude, OpenAI API, Ollama, LiteLLM, Gemini, AI Studio, ChromaDB, Streamlit

System Architecture: Full-Stack Web Applications, API Design, Database Architecture, Deployment Pipelines, Cloud Infrastructure

Cloud & DevOps: Docker, AWS (EC2, S3, ECR), GitHub Actions, CI/CD Pipelines, Infrastructure-as-Code concepts

PROJECTS

Lokr – AI Code Intelligence Engine (Graph-RAG) github.com/Anasmtaweh/lokr Jan 2026 – Feb 2026

Privacy-first code intelligence for local LLM workflows

- Architected a privacy-first Graph-RAG solution to resolve context gaps in local LLM workflows, combining semantic vector search with dependency graph traversal to eliminate retrieval hallucinations.
- Designed a polyglot hybrid retrieval pipeline using Tree-sitter AST parsing, ChromaDB embeddings, and topological context bounding to enhance LLM reasoning accuracy and reduce irrelevant code suggestions.
- Implemented programmatic grounding clamps that constrain LLM outputs to only reference files and functions that actually exist in the dependency graph, preventing hallucinated code suggestions.
- Designed and deployed the engine in 6 days utilizing AI-assisted development, iterating on retrieval strategies based on query performance feedback and real-world validation.
- Extended the system with a Model Context Protocol (MCP) server and designed lokr-assistant, a multi-agent system for code review, fixes, and safety validation before deployment.
- Demonstrated measurable improvement in retrieval precision over naive chunk-based approaches, particularly for cross-file function dependencies and call chain understanding.

**Checkers RL – Reinforcement Learning
Evaluation Framework**

[github.com/Anasmtaweh/Checkers-
ML_D3QN_MCTS](https://github.com/Anasmtaweh/Checkers-ML_D3QN_MCTS)

Nov 2025 – Dec 2025

Comparative RL architecture study with outcome-driven optimization

- Defined clear success metrics upfront (win rate vs. random, head-to-head tournaments) to evaluate learning, structuring all architectural decisions around measurable outcomes rather than proxy metrics.
- Identified and resolved reward hacking where agents optimized for piece captures over actual wins; restructured the incentive landscape to Win=+1.0 / Loss=-1.0, with captures awarded 0.1–0.5 based on jump complexity, ensuring terminal outcomes dominate learning.
- Architected a dual-paradigm comparison: Value-based D3QN (learns Q-values from board state) vs. Search-based AlphaZero with MCTS (combines neural networks with Monte Carlo Tree Search for look-ahead planning). Ran rapid iteration cycles to test and refine architectures empirically.
- Validated algorithmic improvements through extensive head-to-head automated evaluation; AlphaZero achieved 0% loss rate against the strongest D3QN baseline, demonstrating superior performance from planning-aware learning.
- Optimized training pipeline by leveraging Ray for distributed multi-worker asynchronous experience collection, resolving PCIe latency bottleneck (moved small-batch inference to CPU to eliminate GPU-CPU communication overhead), improving iteration throughput from ~2,200s to ~1,750s per iteration.
- Designed and deployed a full-stack Flask Web Arena enabling visual, browser-based benchmarking and human vs. agent sparring to demonstrate final trained policies.

Mishtika – AI-Powered Pet Care Platform pet-ai-render-1.onrender.com

Feb 2025 – May 2025

Full-stack product with AI-driven personalization and workflow automation

- Identified fragmented pet care workflows and designed a centralized full-stack platform to aggregate data, manage schedules, and automate personalized recommendations.
- Architected system with role-based access control, scheduled automation, and production-grade CI/CD pipeline (GitHub Actions → Docker → AWS ECR → EC2). Leveraged test automation to enforce code quality before deployment.
- Improved AI recommendation quality by dynamically injecting structured pet context (species, breed, health history) and localized product datasets into LLM prompts with each user query, enabling highly relevant, context-aware outputs.
- Designed automation patterns using MongoDB scheduling and JWT token expiration to prevent duplicate email notifications across recurring events, and leveraged S3 pre-signed URLs for secure time-limited media access.
- Validated system behavior through formal university evaluation; clearly communicated AI limitations and system design tradeoffs to stakeholders.
- Chose AWS ECR over Docker Hub for image storage (co-located with EC2 target, reduced latency, integrated IAM security, eliminated exposure of credentials in public registries).

EXPERIENCE

Production Operator

Jun 2023 – Aug 2023

Liban Jus, Beirut

- Maintained production workflows and resolved operational issues in a fast-paced manufacturing environment, reinforcing reliability and execution under pressure.

EDUCATION

Bachelor of Science in Computer Science & Information Technology

Graduated Jan 2026

Lebanese International University (LIU), Beirut, Lebanon

Software Development, Artificial Intelligence, Database Systems

CERTIFICATIONS

- AWS Certified Cloud Practitioner (2025–2028)

- Introduction to DevOps – IBM & Coursera