Building Reliable AI Systems:

From Hype to Practical Toolkits

LLM4S

"Let's make AI systems not just smart, but dependable."

Kannupriya Kalra – Engineering Leader @ LLM4S

AI Hype vs Reality

MIT report: 95% of generative AI pilots at companies are failing

BY **SHERYL ESTRADA**
SENIOR WRITER AND AUTHOR OF CFO DAILY

August 18, 2025 at 6:54 AM EDT

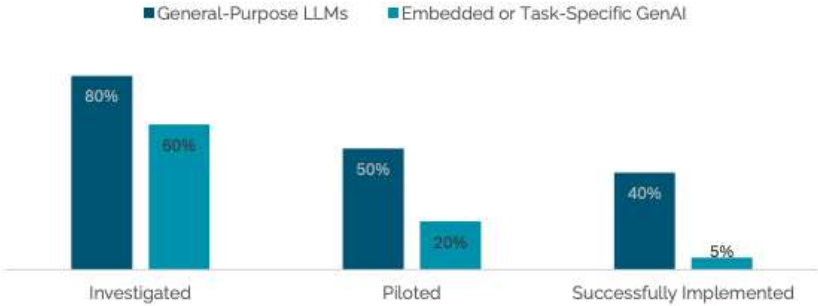
AI hype vs reality: Why technology isn't truly revolutionary as the narrative pushed by tech firms

AI firms have pushed the narrative that AI can replace humans in most jobs, but so far, the technology has done little more than assist programmers and copywriters in their work.

88% of AI pilots fail to reach production — but that's not all on IT

News Analysis
Mar 25, 2025 • 4 mins

Exhibit: The steep drop from pilots to production for task-specific GenAI tools reveals the GenAI divide



AI Hype vs AI Reality: What it takes to get Gen AI into production

By Corey Keyser, Head of AI at Ataccama

LLM4S

AI Hype vs Reality

- Demos amaze, but systems collapse under real load.
- Most GenAI pilots fail before production.
- Key reasons:
  - Hallucination & lack of grounding
  - Scaling & latency scaling issues
  - Management:
    - Observability
    - Measurability
    - Lack of feedback cycles
- From "cool prototypes" → to "critical systems"

LLM4S

From Models to Systems

- **LLM ≠ Product**
- A large language model alone is not a reliable production system.system
- **A Reliable AI System =**
  LLM + Context + Data + Tools + Observability + Governance + Feedback Cycles
- Each piece adds stability, context grounds responses, data ensures truth, tools enable action, observability tracks behavior, governance builds trust, feedback cycles allow us to optimise
- **LLM4S** brings structure and safety to all these moving parts making AI *predictable, scalable, and production-ready.*

LLM4S

The Reliability Imperative

Why Reliability?

- **Trust & Reproducibility** Teams adopt AI only when outputs are consistent and explainable.
- **Governance & Compliance** Organizations need transparency and audit trails to meet safety and regulatory standards.
- **Cost & Performance Control** Reliable systems optimize inference cost, latency, and scalability.
- **Reliability isn’t post-facto testing it’s design.** It’s about building AI like software structured, measurable, and predictable.

LLM4S

Introducing LLM4S

LLM4S – Large Language Models for Scala

A **type-safe, structured, and composable** toolkit for building reliable

What it enables:

⚙️ **Multi-provider integration** OpenAI, Anthropic, Gemini, Ollama, and more.

🔍 **Observability & tracing** Understand, debug, and measure every call.

🔄 **Agentic workflows** Multi-step reasoning and tool calling built in.

🎨 **Multi-modal generation** Text, image, and speech through one unified API.

LLM4S turns LLM experiments into production-grade AI systems.

LLM4S Design Philosophy

Pillars of Reliability in LLM4S

1. **Composable Modules**: RAG, agents, tracing, tools.
2. **Type Safety**: catch prompt and context errors at compile time.
3. **Tracing & Observability**: understand, debug, and optimize.
4. **Multi-provider Support**: OpenAI, Anthropic, Gemini, Ollama.
5. **Scalability**: plug-and-play architectures with consistent APIs.

LLM4S

Retrieval-Augmented Generation (RAG)

Grounded by Design: Retrieval → Generate → Cite

- Connects to multiple data sources: PDFs, CSVs, web pages, and databases.
- Uses FAISS and PGVector for fast, scalable retrieval.
- Ensures factual and context-aware responses — no retrieval means no answer.
- Built-in evaluation for accuracy, latency, cost, and safety.
- Produces outputs that are traceable, measurable, and reliable.

LLM4S

Model Context Protocol (MCP)

One unified protocol for context and tool interaction

- Extends LLM4S agents to connect with any external tool or service.
- Integrates APIs, databases, and cloud systems through a standard interface.
- Enables scaling — tools can be added, swapped, or upgraded without code changes.
- Maintains consistent, modular, and easily manageable AI workflows.

LLM4S

Agentic Workflows

From single prompts to autonomous reasoning

- LLM4S agents can plan, act, and verify through structured multi-step workflows.
- Support execution of commands, API calls, and tool interactions in safe environments.
- Move beyond chat responses to enable reliable automation and reasoning.
- Power diverse use cases — from code generation to enterprise workflows and interactive experiences like *Szork.*

LLM4S

Tracing & Observability

Turning AI’s black box into a transparent system

- LLM4S tracing captures every step — from user input to model response and tool execution.
- Supports OpenTelemetry integrations with tools like Langfuse and MLflow.
- Provides detailed analytics on timelines, token usage, latency, and cost.
- Console tracing enables quick visibility during local development.
- Converts experimentation into measurable performance for debugging and optimization.

LLM4S

Multimodal Capabilities

Unified API for:

- **Image Generation:** Powered by Stable Diffusion for creating dynamic visuals.
- **Speech-to-Text:** Integrates Whisper and CMU Sphinx for accurate voice input.
- **Text-to-Speech:** Uses Tacotron2 for natural and expressive audio output.
- **Tool Calling for Interactivity:** Enables real-time actions in scenarios like inventory, games, and simulations.

LLM4S

Reliability in Production

Production = Predictability

- Safe workspaces (sandboxed agents)
- Retry & error handling built-in
- Config-driven model switching
- Structured logs & metrics
- Works with Scala’s concurrency & parallelism

LLM4S

What You Can Build Today

Use Case	Description	Status
Conversational Agents	Type-safe, multi-turn chatbots	✓✓ Shown
RAG Systems	Enterprise knowledge Q&A	🚧 Coming soon
Code Generation	AI-assisted Scala coding	✓✓ Shown
Image Processing	Multimodal integration	✓✓ Shown
Workflow Automation	Multi-step agents	✓✓ Shown
Semantic Search	Vector retrieval	🚧 In progress

LLM4S

The Road Ahead & Community

Coming Next:

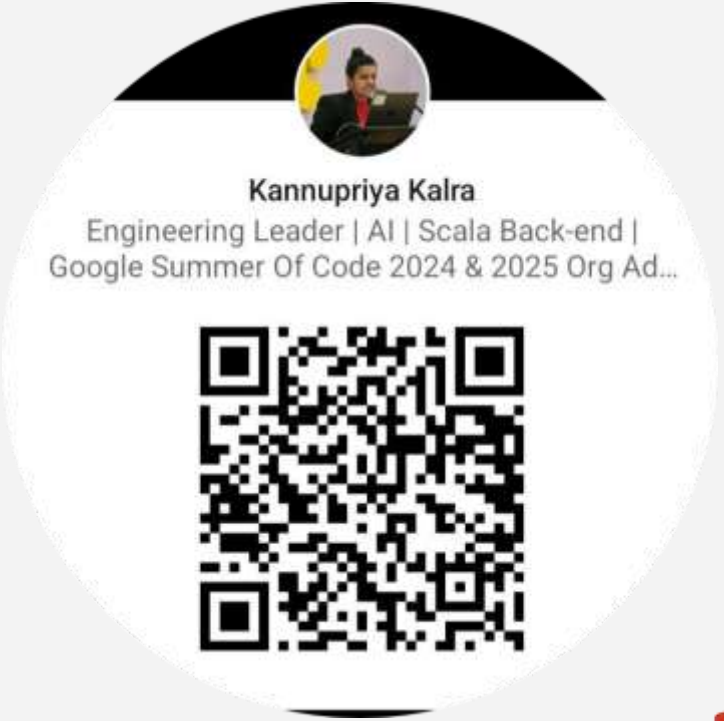
- Streaming responses
- Enhanced MCP introspection
- More model providers
- Better production templates

🌍 **Community:**
100+ global contributors
“Learn AI by Building AI”
🔗 GitHub | Discord | Docs

LLM4S

Key Takeaways & Q&A

- ✅ Reliable AI = Design + Discipline
  ✅ Frameworks > Ad-hoc scripts
  ✅ LLM4S brings structure, reliability & scale
  💬 “Build the cool stuff — and make it *work* in production.”



- 👍 **Contact:** kannupriyakalra@gmail.com
  🔗 linkedin.com/in/kannupriyakalra | github.com/llm4s

LLM4S

Don’t forget to star us on Github :)

LLM4S