

COURSE OUTLINE

Applied AI Solution Architect — FDE

An intensive, lab-driven program for professionals building an AI-enabled full-stack enterprise application

DURATION	FORMAT
5 days	Instructor-led, lab-driven, continuous capstone project
DELIVERY	TRAINER
Online Instructor Led Workshop · Onsite Program for your Teams	Muhammed Shakir

Why this course exists

An AI-enabled enterprise application requires software engineering, Generative AI, retrieval, agentic workflows, full-stack integration, security, responsible AI, governance, and quality.

This program gives participants an end-to-end, build-first understanding of how to architect, design, develop, integrate, deploy, and govern an AI-enabled full-stack enterprise application. Participants build small, inspectable PoC(s) that shall demonstrate the core topics.

Skills You'll Walk Away With

- Apply modern software engineering practices, SDLC, Agile, Git/GitHub, clean code, design principles, design patterns, and code reviews.
- Design prompts, LLM application flows, embeddings and vector search, RAG, advanced RAG knowledge systems, AI agents, and multi-agent enterprise agent architecture.
- Build AI-powered web and conversational applications that connect AI models with enterprise APIs, databases, and enterprise knowledge systems.
- Apply AI-assisted software engineering, AI application architecture, AI UX, human-in-the-loop workflows, AI security, responsible AI, governance, and evaluation.

Who should attend

This program is for technical professionals responsible for AI-enabled enterprise applications.

- **Forward-deployed engineers** — building and integrating AI-enabled full-stack enterprise applications.

- **Solution architects** — defining AI application architecture, application boundaries, technology selection, and enterprise controls.
- **Software engineers** — applying modern software engineering and AI-assisted software engineering to AI-powered applications.
- **Technical leads** — connecting AI application delivery with security, responsible AI, governance, and quality.

Course Outline

Day 1 — Establish the Engineering and LLM Application Foundation

Set up the capstone, define its enterprise AI use case, and make its first LLM application flow work.

Session 1 — Engineering & FDE Fundamentals; AI & Generative AI Foundations

Establish the software engineering and AI foundation for the continuous capstone.

- Modern software engineering practices, SDLC, Agile, Git/GitHub, clean code, design principles, design patterns, and code reviews
- AI/ML overview, Generative AI, LLMs, foundation models, transformers, multimodal AI, and enterprise AI use cases
- *Hands-on:* Establish the capstone as an engineering engagement and define its enterprise AI use case.
 - Create the capstone project in Git/GitHub and define its SDLC and Agile flow.
 - Record the initial application design using clean-code, design-principle, and design-pattern decisions; conduct a focused code review.
 - Map AI/ML, Generative AI, LLM, foundation-model, transformer, and multimodal-AI considerations to the selected enterprise AI use case.
 - **Session checkpoint:** a version-controlled capstone foundation with an explicit engineering/FDE approach and enterprise AI use case.

Session 2 — Prompt Engineering; LLM Application Development

Design the prompt and context approach, then turn it into the capstone's first LLM application flow.

- Prompt design, zero-shot/few-shot prompting, context engineering, structured outputs, and prompt optimization and evaluation
- LLM APIs, model selection, token management, context windows, temperature, structured responses, and API integration
- *Hands-on:* Build the initial prompt-to-LLM application flow.
 - Design zero-shot and few-shot prompts, then define the context-engineering and structured-output approach.
 - Optimize and evaluate the prompts against the intended application behaviour.
 -

Select a model, connect the LLM API, apply token, context-window, and temperature choices, and return structured responses through API integration.

- **Session checkpoint:** a working LLM application flow with documented prompts, context, structured outputs, model selection, token management, context windows, temperature, and API integration.

Day 2 — Build the Knowledge and Agentic AI Foundation

Add retrieval, advanced RAG knowledge capability, and a controlled agentic workflow to the same application.

Session 3 — Embeddings & Vector Search; Retrieval-Augmented Generation – RAG

Make enterprise document content searchable and use it for grounded generation.

- Embeddings, semantic search, vector databases, similarity search, indexing, metadata filtering, and hybrid search
- RAG architecture, document ingestion, chunking, embeddings, retrieval, reranking, grounding, citations, and RAG optimization
- *Hands-on:* Build the capstone's retrieval and RAG path.
 - Create embeddings, index them in a vector database, and implement semantic and similarity search with metadata filtering and hybrid search.
 - Ingest and chunk documents, then implement retrieval and reranking in the RAG architecture.
 - Generate grounded responses with citations and optimize the RAG flow.
 - **Session checkpoint:** a working document-ingestion, vector-search, retrieval, reranking, grounded-generation, and citation path.

Session 4 — Advanced RAG & Knowledge Systems; AI Agents & Agentic AI

Extend RAG into an enterprise knowledge system and add a human-in-the-loop agentic workflow.

- Query transformation, contextual retrieval, multi-source retrieval, knowledge bases, RAG evaluation, and enterprise document intelligence
- AI agents, planning, tool/function calling, memory, reasoning workflows, agent orchestration, and human-in-the-loop
- *Hands-on:* Extend the RAG path and add an AI agent.
 - Add query transformation, contextual retrieval, and multi-source retrieval; organize the content as a knowledge base and evaluate it for enterprise document intelligence.
 - Define an AI agent's planning and reasoning workflow, connect tool/function calling and memory, and orchestrate it with a human-in-the-loop step.
 - **Session checkpoint:** an advanced RAG knowledge system and an AI-agent workflow with visible planning, tools/functions, memory, reasoning, orchestration, and human oversight.

Day 3 — Build the Enterprise Application and Its Integrations

Connect multi-agent capability to the AI-powered application, enterprise APIs, external systems, and enterprise data.

Session 5 — Multi-Agent & Enterprise Agent Architecture; Building AI-Powered Applications

Evolve the agentic workflow into an enterprise agent architecture and expose it through the application experience.

- Multi-agent systems, agent collaboration, enterprise tools/APIs, workflow orchestration, and agent governance
- Integrating LLMs into web applications, AI-enabled UI/UX, chat interfaces, conversational applications, and AI workflows
- *Hands-on:* Build the AI-powered enterprise application experience.
 - Define multi-agent collaboration, connect enterprise tools/APIs, orchestrate the workflow, and record the agent-governance approach.
 - Integrate the LLM capability into the web application and create the AI-enabled UI/UX, chat interface, conversational application, and AI workflows.
 - **Session checkpoint:** an AI-powered web and conversational application with multi-agent collaboration, enterprise tools/APIs, workflow orchestration, and agent governance.

Session 6 — AI + API Engineering; AI + Database Integration

Connect the AI-powered application with enterprise APIs, external systems, and enterprise data.

- Connecting AI models with enterprise APIs, tool calling, function calling, API orchestration, and external system integration
- Structured and unstructured data, vector databases, SQL generation, natural-language data access, and enterprise knowledge systems
- *Hands-on:* Implement enterprise API and database integration in the capstone.
 - Connect AI models with enterprise APIs, add tool calling and function calling, and orchestrate APIs for external system integration.
 - Connect structured and unstructured data, vector databases, SQL generation, natural-language data access, and enterprise knowledge systems to the application.
 - **Session checkpoint:** AI-to-enterprise-API and AI-to-database integration across external systems, structured and unstructured data, vector databases, SQL generation, natural-language data access, and enterprise knowledge systems.

Day 4 — Refine the Application Architecture and Human Experience

Use AI-assisted software engineering to improve the capstone, make its architecture explicit, and add AI UX, human-in-the-loop, and security controls.

Session 7 — AI-Assisted Software Engineering; AI Application Architecture

Refine the capstone and consolidate its architecture.

- GitHub Copilot/AI coding assistants, code generation, debugging, refactoring, test generation, documentation, and developer productivity
- AI application patterns, LLM orchestration, RAG architecture, agent architecture, application boundaries, and technology selection
- *Hands-on:* Improve and architect the integrated solution.
 - Apply GitHub Copilot/AI coding assistants for code generation, debugging, refactoring, test generation, and documentation; review developer productivity.
 - Map the AI application patterns, LLM orchestration, RAG architecture, and agent architecture; define application boundaries and technology selection.
 - **Session checkpoint:** a refined capstone with explicit AI application patterns, LLM/RAG/agent architecture, application boundaries, and technology selection.

Session 8 — AI UX & Human-in-the-Loop; AI Security

Add human oversight to the application experience and address AI application security risks.

- Conversational UX, explainability, human validation, feedback mechanisms, escalation, and approval workflows
- Prompt injection, jailbreaks, data leakage, insecure tool use, model abuse, adversarial inputs, and AI application security
- *Hands-on:* Complete the AI UX and human-in-the-loop flow, then apply AI security.
 - Refine conversational UX and explainability; add human validation, feedback mechanisms, escalation, and approval workflows.
 - Identify prompt-injection, jailbreak, data-leakage, insecure-tool-use, model-abuse, and adversarial-input risks; apply AI application security.
 - **Session checkpoint:** an AI UX with human oversight and an explicit AI application security approach.

Day 5 — Govern and Evaluate the Completed Enterprise Application

Apply responsible AI, governance, and quality evaluation to the completed capstone.

Session 9 — Responsible AI; AI Governance

Establish responsible-AI practices, governance, and enterprise controls.

- Fairness, transparency, explainability, privacy, safety, hallucination management, and human oversight
- AI risk management, model governance, auditability, data governance, compliance, policies, and enterprise controls
- *Hands-on:* Apply responsible AI and governance to the capstone.

- Review fairness, transparency, explainability, privacy, safety, hallucination management, and human oversight in the application.
- Define AI risk management, model governance, auditability, data governance, compliance, policies, and enterprise controls.
- **Session checkpoint:** responsible-AI practices and an AI-governance approach integrated into the capstone.

Session 10 — AI Evaluation & Quality

Evaluate the completed AI-enabled enterprise application.

- Accuracy, relevance, groundedness, hallucination measurement, robustness, toxicity, bias, and LLM evaluation
- *Hands-on:* Evaluate the completed capstone and consolidate its end-to-end outcome.
 - Evaluate accuracy, relevance, groundedness, hallucination measurement, robustness, toxicity, and bias through LLM evaluation.
 - Review the completed application from software engineering through AI, RAG, agentic AI, full-stack integration, security, responsible AI, governance, and quality.
 - **Session checkpoint:** a completed AI-enabled full-stack enterprise application with its AI evaluation and quality approach made visible.

Capstone Project

Participants incrementally build one **AI-enabled full-stack enterprise application**. Day 1 establishes the software engineering, FDE, AI/Generative AI, prompt-engineering, and LLM-application foundation. Day 2 adds embeddings and vector search, RAG, advanced RAG and knowledge systems, and agentic AI. Day 3 turns these capabilities into an AI-powered application with multi-agent architecture, enterprise API integration, and enterprise database integration. Day 4 refines the application through AI-assisted software engineering, makes its architecture explicit, and adds AI UX, human-in-the-loop, and AI security. Day 5 applies responsible AI, AI governance, and AI evaluation and quality.

The application builds in one direction. Each session extends the same capstone; participants do not replace it with a new demonstration. The final application makes AI application patterns, LLM orchestration, RAG architecture, agent architecture, enterprise tools/APIs, enterprise knowledge systems, human oversight, security, governance, and evaluation visible in the same solution.

In scope: modern software engineering, AI and Generative AI, prompt engineering, LLM application development, embeddings and vector search, RAG, advanced RAG and knowledge systems, AI agents and agentic AI, multi-agent and enterprise agent architecture, AI-enabled full-stack development, AI security, responsible AI, AI governance, and AI evaluation and quality.

A Note on Delivery

The five-day sequence is deliberate: establish the engineering and LLM application foundation; build knowledge and agentic capability; create the enterprise application and its integrations; refine its architecture, human experience, and security; then govern and evaluate the completed solution. Each day contains two sessions, and each session balances explanation, demonstration, guided hands-on work, inspection, and a capstone checkpoint.

Ready to architect and build AI-enabled enterprise applications?

Enquire about a private cohort or a tailored delivery for your team.