ARTIFICIAL INTELLIGENCE | ADVANCED | 5 DAYS

Advanced Agentic AI for Enterprise Bootcamp

Design, build, and deploy enterprise-ready Agentic AI systems. This bootcamp focuses on multi-agent orchestration, autonomous workflows, enterprise RAG, observability, governance, and deployment. Teams learn how to move from experimental LLM apps to auditable, secure, and compliant Agentic AI solutions aligned with real business workflows.

Why Agentic AI?

Enterprise-ready autonomy

Move from static chatbots to multi-agent systems that reason, plan, and act.

Governed AI adoption

Build auditable, traceable, and policy-aligned AI agents.

Operational efficiency

Automate complex workflows across finance, compliance, and operations.

Future-proof architecture

Adopt emerging standards like MCP and ACP for cross-tool interoperability.

What you'll be able to do

- ✓ Understand enterprise multi-agent architecture patterns and governance-first design principles.
- ✓ Build tool-using agents with safe input/output contracts, state handling, and retry-based execution patterns.
- ✓ Implement enterprise-grade agents using LangChain, DSPy, LangGraph, and CrewAI for reliable orchestration and collaboration.
- ✓ Design and build production-ready RAG pipelines with traceable citations, source metadata, and audit trails.
- ✓ Apply RAG quality controls including permission-aware retrieval, document freshness strategies, and hallucination reduction guardrails.
- ✓ Implement planning and reflection loops with safety guardrails including tool allowlists and prompt injection defenses.
- ✓ Add observability and compliance-ready logging with tracing, PII-aware practices, dashboards, and alerting.
- ✓ Build evaluation and testing frameworks for agent systems including regression testing and red-team style validations.
- ✓ Deploy enterprise agent systems using secure access controls, tenancy boundaries, quotas, rate limiting, and secrets management.
- ✓ Deliver a capstone production-style enterprise multi-agent solution with architecture documentation, governance controls, and an end-to-end working demo.

Who this is for

- AI and ML Engineers
- Data Scientists and MLOps Engineers
- Platform and DevOps Engineers
- Product Managers working on AI products
- Enterprise Architects and Tech Leads

Curriculum

01 Module 1: Enterprise Multi-Agent Architecture (Governance-First Design)

- What multi-agent systems solve in enterprises (automation, decision support, ops copilots)
- Key roles in an agent system (planner, executor, verifier, reviewer)
- Enterprise governance requirements (auditability, traceability, access control)
- Design patterns (orchestrator-agent, swarm, supervisor-worker)
- **Hands-on:** Activity: Design a multi-agent system blueprint for an enterprise workflow

02 Module 2: Tool-Using Agents Fundamentals (Function Calling + Tooling Patterns)

- Agent tool use patterns (search, retrieval, actions, workflows)
- Tool schemas and safe input/output contracts
- State management basics (memory, context boundaries)
- Error handling and retries for tool execution
- **Hands-on:** Lab: Build a simple tool-using agent that calls 2 tools and returns structured output

03 Module 3: LangChain Agents (Enterprise-Ready Implementation)

- LangChain fundamentals (chains, tools, agents)
- Runnable patterns and structured outputs
- Prompt templates and role-based constraints
- Secure tool execution and sandboxing concepts
- **Hands-on:** Lab: Implement a LangChain agent that performs multi-step task execution with tool calls

04 Module 4: DSPy for Reliable Agent Behavior (Prompt Optimization Patterns)

- Why DSPy (reduce prompt brittleness, improve consistency)
- Signatures, modules, and programmatic prompting
- Optimization loop overview (teleprompters, metrics)
- Evaluation-driven improvement for agent outputs
- **Hands-on:** Lab: Build a DSPy program to improve answer quality and reduce hallucinations

05 Module 5: LangGraph for Stateful Multi-Step Agents (Orchestration)

- Graph-based agents (nodes, edges, state, transitions)
- Handling branching, retries, and recovery paths
- Human-in-the-loop checkpoints (approval gates)
- Building reliable workflows (idempotency and re-entrancy concept)
- **Hands-on:** Lab: Create a LangGraph workflow with planner → executor → verifier loop

06 Module 6: CrewAI for Multi-Agent Collaboration (Role-Based Teams)

- CrewAI concepts (agents, tasks, crew orchestration)
- Defining roles and responsibilities for enterprise use
- Task decomposition and parallel execution patterns
- Managing shared context and preventing cross-agent contamination
- **Hands-on:** Lab: Build a CrewAI team with 3 agents (researcher, builder, reviewer) for an enterprise task

07 Module 7: Enterprise RAG Design (Traceable Citations + Audit Trails)

- RAG fundamentals (chunking, embeddings, retrieval, re-ranking)
- Enterprise RAG requirements (permissioning, tenancy, governance)
- Traceable citations and grounded responses
- Audit trails (who asked what, what sources were used, when)
- **Hands-on:** Lab: Build a RAG pipeline that returns answers with citations and source metadata

08 Module 8: RAG Quality Controls (Freshness, Access, and Hallucination Reduction)

- Reducing hallucinations with strict grounding rules
- Document freshness and versioning strategy
- Permission-aware retrieval (RBAC/ABAC concepts)
- Fallback strategies (no answer, escalate, human review)
- **Hands-on:** Lab: Add policy checks and refusal logic when documents are missing or restricted

09 Module 9: Planning, Reflection, and Safety Guardrails

- Planning strategies (task decomposition, step planning, goal validation)
- Reflection loops (self-checking, verifier agent pattern)
- Safety guardrails (tool allowlists, restricted actions, scope limits)
- Prompt injection defenses for RAG and tool use
- **Hands-on:** Lab: Implement planner + verifier loop with safety filters and prompt injection checks

10 Module 10: Observability and Compliance-Ready Logging

- What to log for enterprise compliance (inputs, outputs, tool calls, sources)
- Tracing agent steps and decisions (run-level telemetry)
- PII redaction and data minimization concepts
- Operational dashboards and alerting for agent failures
- **Hands-on:** Lab: Add structured logs + traces for agent runs and generate an audit report

11 Module 11: Evaluation and Testing for Agent Systems

- Offline vs online evaluations (test suites, golden datasets)
- Metrics (groundedness, citation accuracy, task success rate)
- Regression testing for prompts and tools
- Red-team testing basics (jailbreaks, injection, data leakage)
- **Hands-on:** Lab: Build an evaluation harness and run tests against agent changes

12 Module 12: Deployment Patterns for Enterprise Agents

- Deployment models (internal API, chatbot UI, workflow integration)
- Access controls and tenancy boundaries
- Rate limiting, quotas, and cost controls
- Secure secrets handling and key management
- **Hands-on:** Lab: Package the agent system as a service with environment configs and safe runtime controls

13 Module 13: Capstone Project (Production-Style Enterprise Multi-Agent System)

- Capstone goal: Deliver a governance-aligned multi-agent solution
- Implement tool-using agent team (LangChain/LangGraph/CrewAI)
- Add enterprise RAG with citations and audit logging
- Implement planning + reflection + safety guardrails
- Deliver observability, evaluation, and compliance-ready reporting
- **Hands-on:** Capstone: Submit architecture diagram, governance controls, and end-to-end working demo