



AI Agent Development: Building LLM Agents with Tools and RAG



AI Agent Development: Building LLM Agents with Tools and RAG

Introduction:

Many organisations have tried large language model pilots that answer questions well in a demo but fail when asked to look up records, call business systems or cite internal documents. AI agent development closes that gap: engineering agents that choose tools, retrieve the right evidence and finish multi-step tasks reliably. This Core Concept course takes developers through agent architectures, tool calling, retrieval-augmented generation RAG, memory, testing and deployment in hands-on labs. Participants finish with a Business Agent Build Pack: a working agent plus its architecture, tool specifications, test results and running costs.

Course Objectives:

- Choose between a fixed workflow, a single agent and a multi-agent design for a given business task, and justify the choice with expected accuracy, cost and maintenance effort
- Implement tool and function calling so an agent reads from and writes to business systems through typed, well-described interfaces
- Build a retrieval-augmented generation RAG pipeline with embeddings, vector search and reranking that grounds agent answers in company documents
- Design conversation memory and task state so an agent keeps context across steps and sessions
- Measure agent quality with task success tests and hallucination checks, and add code-level guardrails, cost limits and latency targets
- Deploy and monitor a business agent and document it in a Business Agent Build Pack

Target Audience:

- Software developers who integrate large language model APIs into applications and internal tools
 - Data engineers who prepare document stores, embeddings and pipelines that feed AI applications
 - Automation specialists who move rule-based process automation towards language-model-driven agents
 - Machine learning practitioners who need to turn model prototypes into tool-using agents
 - Solution builders who compare low-code agent platforms with code-first development for their teams
-

Course Outline:

Day 1: LLM Agent Foundations and the Build Environment

- Workflows Versus Agents: Predefined Code Paths and Model-Directed Control
- The Agent Loop: Model Call, Tool Selection, Observation and Stopping Rule
- Python Development Environment, API Keys and Model Endpoint Set-Up
- Candidate Business Task Inventory Scored by Volume, Rules and Data Access
- Low-Code Agent Builders Compared with Code-First Development

Day 2: Agent Architecture Patterns and Tool Interfaces

- Prompt Chaining and Routing Workflow Patterns
- Parallelisation by Sectioning and Voting
- Orchestrator-Workers and Planner-Executor Multi-Agent Designs
- Evaluator-Optimizer Loops for Self-Checking Output
- Agent-Computer Interface Design: Tool Names, Parameter Descriptions and Usage Examples

Day 3: Tool Calling, Retrieval and Memory in Code

- Function Calling with Typed JSON Arguments and Response Parsing
- Embedding Models and Vector Database Similarity Search
- Chunking Strategies: Fixed-Length Overlap, Sentence Splitting and Format-Aware Splitting
- Reranking Retrieved Passages and Grounded Answer Assembly
- Conversation Memory, Session State and Long-Term Store Design

Day 4: Evaluation, Guardrails, Cost and Operations

- Agent Test Harness: Golden Task Sets and Task Success Rate
- Hallucination Checks: Groundedness Scoring Against Retrieved Sources
- Guardrails in Code: Input Validation, Output Schema Enforcement and Retry Logic
- Token Budgets, Model Tiering, Response Caching and Latency Profiling
- Containerised Deployment, Tracing, Structured Logging and Monitoring Dashboards

Day 5: Business Agent Lab Build and Build Pack

- Lab Build: Customer Enquiry Triage Agent with Ticketing Tool Calls
 - Lab Build: Internal Policy Question-Answering Agent over a Document Store
 - Evaluation Run and Failure Trace Review on the Lab Agents
 - Business Agent Build Pack Drafting: Architecture, Tool Specifications, Test Results and Run Costs
 - Peer Code Review and Build Pack Walkthrough
-

Skills You Will Gain:

- Agent Architecture Selection
- Tool Interface Engineering
- Retrieval Pipeline Development
- Vector Search Tuning
- Agent State Management
- Agent Test Automation
- Inference Cost Optimisation
- Agent Observability

Why Attend This Course:

- Leave with a Business Agent Build Pack and working agent code that colleagues can run, test and extend
- Avoid over-engineering by knowing when a simple workflow beats an autonomous agent for a given task
- Diagnose why an agent picks the wrong tool, retrieves the wrong passage or loops, using traces instead of guesswork
- Compare build approaches with developers from banking, retail, logistics, energy and public services

Conclusion:

Agents earn their place in production only when their tools, retrieval and memory are engineered deliberately and their behaviour is measured rather than assumed. This course moves from the agent loop and the build environment, through workflow and multi-agent patterns, to tool calling, RAG pipelines and memory written in code, and then to testing, guardrails, cost control and monitoring. The final day applies everything in two lab builds and produces a Business Agent Build Pack ready for technical review and further development.
