JUPITER PRIME

Consulting • Training • Advisory

جوبيتر برايم للاستشارات والتدريب

COURSE BROCHURE

AI Agent Design and Development Training Course

28 June – 2 July 2027
New York

jupiterprime.uk

info@jupiterprime.uk

Registered in the United Kingdom and Saudi Arabia

AI Agent Design and Development Training Course

Ref: 72_1826 **Date:** 28 June – 2 July 2027 **Location:** New York **Fees:** 9500 GBP

AI Agent Design and Development is a 1-week training course for product managers, solution architects, developers and automation leads, who leave with a working AI agent design specification and its evaluation plan. Language models can now plan steps, call tools and act on systems, so poorly scoped agents can trigger real transactions and real damage. Jupiter Prime Training runs this course on AI agent design and development for those who build and own these systems.

Why AI Agent Design and Development Matters Now

Three forces shape AI agent design and development today: model providers adding tool calling and long context that make autonomous task chains practical, business owners asking for agents that close tickets rather than merely answer questions, and security and risk teams worried about agents with write access to live systems. Teams that design scope, tools and checks carefully ship agents that save hours and fail safely; teams that skip design ship agents that loop, leak data or take the wrong action. Examples come from telecommunications customer operations and insurance claims handling. Controls draw on ISO/IEC 42001 for AI management systems and the NIST AI Risk Management Framework AI RMF.

Who Gains Most from AI Agent Design and Development

- Product managers who define agent goals, user journeys and success measures
- Solution architects responsible for integration, permissions and system boundaries
- Developers who build tool interfaces, prompts, memory stores and orchestration code
- Automation leads who decide which workflows move from scripts to agents
- Quality and security engineers who test agent behaviour before release

The strongest gain comes when a product manager, an architect and a developer building the same agent attend together.

AI Agent Design and Development Capabilities

Participants learn to turn a business task into an agent with clear goals, bounded tools, useful memory and measurable behaviour.

- Apply task decomposition to split a business goal into agent steps — product owners
- Design tool schemas and permission scopes for safe system actions — solution architects
- Specify short-term memory, long-term memory and retrieval sources — development leads
- Weigh single-agent and multi-agent orchestration patterns for a workflow — architecture boards
- Test agents with evaluation sets, trajectories and failure injection — release managers
-

Build an AI agent design specification with guardrails and an evaluation plan — delivery sponsors

Key AI Agent Design and Development Terms

- AI Agent — a system in which a language model plans steps, calls tools and acts toward a goal with limited human input.
- Tool Calling — the ability of a model to select a defined function or interface and pass it structured arguments.
- Agent Memory — stored context an agent keeps within a task or across sessions to stay consistent and informed.
- Retrieval-Augmented Generation — fetching relevant documents or records at run time so the agent grounds its reasoning in approved sources.
- Multi-Agent Orchestration — coordinating several specialised agents through a planner, router or shared state to complete larger tasks.
- Agent Evaluation — testing an agent's final answers and intermediate steps against expected behaviour, cost and safety criteria.

AI Agent Design and Development Day by Day

Day 1: From Chatbots to Agents

- Agent Architecture of Model, Tools, Memory and Loop — Design Boundaries
- Workflow Candidates for Agents — Automation, Assistance or Autonomy
- Task Decomposition and Planning Patterns — Step and Goal Definition
- Human-in-the-Loop Checkpoints — Approval Before Consequential Actions
- Success Measures for Agents — Completion Rate, Cost and Time

Day 2: Tools, Actions and Integration

- Tool Schema Design and Function Descriptions — Reliable Model Selection
- Permission Scopes and Read Versus Write Access — Blast Radius Limits
- API Integration With Business Systems — Error Handling and Retries
- Prompt and System Instruction Design for Agents — Behaviour Consistency
- Logging and Trace Capture — Audit of Every Agent Step

Day 3: Memory, Retrieval and Context

- Short-Term Context and Long-Term Memory Stores — What to Remember
- Retrieval-Augmented Generation Pipelines — Chunking, Embeddings and Ranking
- Knowledge Freshness and Source Control — Grounding Quality Decisions
- Context Window Budgeting — Cost and Latency Trade-Offs
- Personal Data in Memory — Retention and Deletion Rules

Day 4: Orchestration, Evaluation and Guardrails

- Planner, Router and Supervisor Patterns — Multi-Agent Coordination Choices
- Evaluation Sets and Trajectory Scoring — Release Readiness Gates
- Prompt Injection, Tool Misuse and Looping Tests — Failure Containment
- Output Validators and Policy Guardrails — Blocking Unsafe Actions
- ISO/IEC 42001 and NIST AI RMF Controls — Agent Accountability

Day 5: Practising the AI Agent Design Specification

- Use Case Brief and Goal Statement — Applying Task Decomposition
- Tool Catalogue and Permission Map — Applying Integration Design
- Memory and Retrieval Design — Applying Context Methods
- Evaluation Plan and Guardrail Set — Applying Testing Methods
- AI Agent Design Specification — Peer Review and Sign-Off

Applying AI Agent Design and Development After the Course

- Write tool schemas and permission scopes for one internal workflow and review them with security
- Build a twenty-case evaluation set for an existing assistant and score its trajectories before extending its tools

AI Agent Design and Development Questions FAQ

What is AI agent design and development and why does it matter now?

AI agent design and development is the practice of building language-model systems that plan, call tools and act toward goals. It matters now because agents can change live records and transactions, so scope, permissions and testing decide whether they help or harm.

Which roles gain most from AI agent design and development training?

Product managers, solution architects, developers, automation leads and quality engineers gain most. They choose what an agent may do, connect it to systems and prove it behaves before release.

How does AI agent design and development training differ from a prompt engineering course?

It goes beyond single prompts to whole agent systems. You design tools, permissions, memory, orchestration and evaluation, rather than refining the wording of individual requests to a chat assistant.

What do you bring back from AI agent design and development training?

You bring back a design specification for one agent, with its goal, tool catalogue, permission map, memory design, guardrails and evaluation plan, ready for peer review and build.

AI Agent Design and Development: The Way Forward

The AI agent design specification first informs whether the chosen workflow is built as an assistant with approvals or as a bounded autonomous agent. Product managers, architects and developers use it to agree tools, permissions and release gates. After the first weeks in use, revisit the evaluation set, since real requests expose paths no test case anticipated.