

Course description

Design, build, evaluate, deploy, and secure production-ready agentic AI solutions on Google Cloud. Gain hands-on experience with custom agents, multi-agent orchestration, Agent Development Kit (ADK), Model Context Protocol (MCP), Agent2Agent (A2A), RAG, agent memory, evaluation, deployment, security, and governance.

Learners will progress from configuring low-code agentic workflows and coding agents to building custom agents using the Agent Development Kit, integrating enterprise knowledge through RAG and vector retrieval, implementing agent memory and identity, and orchestrating multi-agent workflows using protocols such as MCP and A2A.

The course also covers the complete production lifecycle of agentic applications, including evaluation, deployment, scaling, observability, troubleshooting, security, governance, guardrails, and human-in-the-loop controls.

The curriculum is aligned with the Professional Agentic Architect certification exam domains and the tools identified as being in scope in the official exam guide.

- Course level: Professional / Advanced
- Duration: 5 days

Activities

This course includes instructor-led sessions, hands-on demonstrations, and interactive discussions to reinforce learning.

Course objectives

In this course, you will learn to:

- Design and configure agentic workflows using low-code tools.
- Create effective system instructions and prompt templates.
- Connect agents securely to enterprise data sources.
- Configure coding agents with MCP servers, Skills, Extensions, Hooks, and tools.
- Use coding agents for code generation, refactoring, optimization, and vulnerability remediation.
- Build custom agents using Agent Development Kit (ADK).
- Configure sessions and agent memory.
- Design RAG and vector retrieval solutions.
- Configure agent identity and permissions.
- Integrate agents with APIs, databases, SaaS applications, and external tools.
- Orchestrate multi-agent workflows using MCP and A2A.
- Implement sequential, parallel, and graph-based agent workflows.
- Build evaluation datasets and continuous agent evaluation pipelines.
- Deploy and scale production agentic workloads.
- Monitor and troubleshoot agent performance.
- Implement authentication, secure tool execution, and access boundaries.
- Apply governance, guardrails, Model Armor, Agent Gateway, and human-in-the-loop controls.

Key features

- Enterprise Agent Architecture: Design production-ready autonomous AI solutions.
- Custom Agent Development: Build agents using ADK and modern agent development patterns.

- Coding Agents: Configure and customize coding agents using MCP, Skills, Extensions, Hooks, and subagents.
- RAG & Enterprise Knowledge: Connect agents to enterprise data using retrieval and vector search.
- Multi-Agent Architecture: Implement MCP and A2A-based agent collaboration.
- Agent Evaluation: Create golden datasets, evaluation sets, and continuous evaluation workflows.
- Production Deployment: Deploy and scale agents using Agent Runtime, Cloud Run, and GKE.
- Security & Governance: Implement Agent Identity, Agent Gateway, Model Armor, OAuth 2.0, policies, and guardrails.
- Hands-On Learning: Practical labs and architecture scenarios throughout the course.
- Capstone Project: Design and implement a production-ready multi-agent enterprise solution.

Intended audience

This course is intended for:

- AI/ML Engineers
- Generative AI Engineers
- AI Architects
- Cloud Architects
- Solution Architects
- Software Architects
- Application Developers
- ML Engineers
- DevOps Engineers
- Platform Engineers
- Technical Leads
- AI Solution Designers
- Professionals preparing for the Professional Agentic Architect certification

Prerequisites

We recommend that attendees of this course have:

- Strong programming fundamentals.
- Experience with Python or another programming language.
- Familiarity with APIs and JSON.
- Basic understanding of LLMs and generative AI.
- Familiarity with cloud concepts.
- Experience with Git and software development workflows.
- Basic understanding of databases and data retrieval is recommended.
- Prior exposure to AI agents is beneficial but not mandatory.

Products covered

- Agent Development Kit (ADK)
- Agent evaluation
- Agent Gateway
- Agent Identity
- Agent Registry
- Agent Retrieval
- Vector Search
- Agent Runtime
- Agent Search
- Agents CLI
- Antigravity
- MCP
- A2A
- Gemini LLMs
- Model Garden
- RAG Engine
- Model Armor
- Sensitive Data Protection

- Cloud Run
- Google Kubernetes Engine
- BigQuery
- Cloud SQL
- Cloud Storage
- Firestore
- Memorystore
- Google Cloud Observability These technologies are drawn from the official exam guide's in-scope tools list.

Course outline

Introduction to Agentic AI & Low-Code Agent Development

- Introduction to agentic AI
- Agentic workflows and autonomous systems
- Agent architecture fundamentals
- Agent states and transitions
- State-based workflows
- Pages and transition routes
- Event handlers
- Configuring agents using low-code tools
- System instructions
- Prompt templates
- Few-shot prompting
- Agent behavior configuration
- Enterprise agent use cases The exam guide places low-code agentic workflow configuration and enterprise data connectivity under Section 1. Learning Objectives
- Explain core concepts of agentic AI.
- Design state-based agent workflows.
- Configure agent behavior using system instructions.
- Create prompt templates for agent workflows.
- Connect agents to enterprise information.

Coding Agents & Gemini CLI/Agent Development

- Introduction to coding agents
- Coding agents vs traditional AI assistants
- Configuring coding agents
- MCP servers
- Agent Skills
- Tools and tool permissions
- Extensions and plugins
- Hooks and rules
- Subagents
- Secure coding environments
- Coding agents in sandboxed environments
- Code generation
- Code refactoring
- Runtime optimization
- Application vulnerability remediation
- Coding-agent best practices The source specifically identifies MCP servers, custom Skills, tools, secure sandboxes, source-code refactoring, runtime optimization, and application-layer vulnerability patching as key coding-agent capabilities. Learning Objectives
- Configure coding agents for software development.
- Extend coding agents using MCP and Skills.
- Use coding agents to analyze and modify code.
- Implement secure coding-agent workflows.
- Automate development and remediation activities.

Customizing Coding Agents for Enterprise Workflows

- Enterprise coding-agent architecture
- Creating Skills
- Plugins
- Extensions
- Hooks

- Rules
- Subagents
- Custom tools
- Agents CLI
- Agent vs human mode
- Standardizing developer workflows
- Enterprise coding workflows
- Scaling coding agents
- Governance of coding agents
- Optimizing deployed agents The exam guide specifically includes creating Skills, plugins, Extensions, Hooks, rules, and subagents and augmenting coding agents using Agents CLI. Learning Objectives
- Customize coding agents for enterprise development.
- Build reusable Skills and Extensions.
- Configure Hooks and rules.
- Create specialized subagents.
- Govern and optimize coding-agent workflows.

Custom Agent Development with ADK

- Agent Development Kit overview
- Agent architecture
- Creating custom agents
- Agent instructions
- Tools
- Tool calling
- Agent execution
- Sessions
- Context
- Agent memory
- Managed sessions
- Memory Bank
- State management

- Agent lifecycle
- Model selection
- LLM vs SLM
- Open-source vs proprietary models
- SaaS vs self-hosted models
- Model selection based on cost, security, and performance The official guide identifies ADK, sessions, Memory Bank, managed sessions, and model-selection considerations within custom-agent development. Learning Objectives
- Build custom agents using ADK.
- Select appropriate models for agent workloads.
- Configure sessions and memory.
- Manage agent state and context.
- Design agents according to cost, security, and performance requirements.

Enterprise Knowledge, RAG & Agent Retrieval

- Enterprise knowledge integration
- Retrieval-Augmented Generation
- RAG architecture
- Document ingestion
- Chunking
- Embeddings
- Vector retrieval
- Similarity scoring
- Reranking
- Vector databases
- Agent Retrieval
- Vector Search
- RAG Engine
- Multimodal enterprise data
- Data access permissions
- Grounded agent responses

• RAG troubleshooting The exam guide specifically calls out RAG pipelines, embeddings, similarity scoring, reranking, Vector Search, and Agent Retrieval. Learning Objectives

• Design enterprise RAG architectures.

• Configure vector-based retrieval.

• Select appropriate retrieval strategies.

• Improve agent grounding and response quality.

• Integrate enterprise knowledge securely.

Agent Tools, Identity & Enterprise Integrations

• Agent tools and capabilities

• Agent Identity

• Agent permissions

• Identity propagation

• Agent Registry

• Google Cloud MCP Servers

• Custom integration layers

• Database integrations

• API integrations

• SaaS integrations

• Remote MCP servers

• Secure agent-to-tool communication

• OAuth 2.0

• Tool authorization

• Least-privilege access Learning Objectives

• Configure secure agent identities.

• Control agent permissions.

• Integrate agents with enterprise applications.

• Connect agents to APIs and databases.

• Secure agent-to-tool communication.

Multi-Agent Systems & Agentic Orchestration

• Multi-agent architecture

- Agent specialization
- Agent handoffs
- Agent2Agent protocol
- MCP-based agent interactions
- Sequential agents
- Parallel agents
- Graph workflows
- Agent coordination
- Agent policies
- Agent Registry
- Agent Runtime
- Agent Identity
- Multi-agent failure handling
- Agent communication patterns The source specifically identifies MCP and A2A, along with sequential, parallel, and graph workflows, as core agent orchestration concepts. Learning Objectives
- Design multi-agent architectures.
- Implement agent handoffs.
- Select sequential, parallel, or graph-based workflows.
- Coordinate specialized agents.
- Apply policies to multi-agent systems.

Agent Evaluation & Testing

- Why agent evaluation is different
- Agent evaluation lifecycle
- Test sets
- Golden datasets
- Prompts and expected responses
- Edge cases
- Tool execution evaluation
- Retrieval evaluation
- Response quality

- Custom autoraters
- ADK evaluation
- Evaluation services
- Continuous evaluation
- Regression testing
- Agent quality metrics
- Evaluation troubleshooting The exam guide specifically includes golden data, prompts, edge cases, continuous evaluation, ADK evaluation tooling, evaluation services, and custom autoraters. Learning Objectives
- Create agent evaluation datasets.
- Evaluate agent responses and tool execution.
- Measure retrieval quality.
- Build continuous evaluation pipelines.
- Identify agent quality regressions.

Deploying, Scaling & Observing Agentic Workloads

- Production agent architecture
- Agent Runtime
- Cloud Run
- GKE
- Deployment strategies
- Scaling agents
- Runtime selection
- Performance optimization
- Latency optimization
- Tool invocation latency
- Agent reasoning loops
- Agent drift
- System failures
- Cloud Logging
- Cloud Trace
- Agent observability

- Cost optimization
- Production troubleshooting The source identifies Agent Runtime, Cloud Run, and GKE as deployment options and specifically calls out troubleshooting drift, tool latency, reasoning loops, system failures, hallucinations, and performance bottlenecks. Learning Objectives
- Select an appropriate agent runtime.
- Deploy and scale production agents.
- Monitor agent performance.
- Troubleshoot agent failures.
- Optimize latency, reliability, and cost.

Security, Governance & Responsible Agentic AI

- Agent security architecture
- Authentication
- OAuth 2.0
- Secure tool execution
- Agent-to-tool authentication
- Agent Identity
- Principal Access Boundary
- Agent Gateway
- Agent Registry
- Model Armor
- Data protection
- Identity propagation
- Agent policies
- Safety frameworks
- Guardrails
- Human-in-the-loop
- Prompt injection
- Data exfiltration
- Sensitive information protection
- Secure agent behavior

• Governance and compliance The source identifies authentication, OAuth 2.0, Principal Access Boundary policies, Agent Gateway, Agent Registry, Model Armor, guardrails, HITL, and secure data/identity propagation as core security and governance capabilities. Learning Objectives

• Design secure agentic architectures.

• Implement authentication and authorization.

• Control agent permissions.

• Protect enterprise data.

• Implement guardrails and safety controls.

• Configure governance policies.

• Incorporate human approval into high-risk workflows.

Capstone - Enterprise Multi-Agent Architecture

• Custom ADK agents

• Gemini model

• Agent sessions and memory

• RAG / enterprise knowledge

• Vector retrieval

• Agent Identity

• MCP integration

• A2A-based agent communication

• Multi-agent orchestration

• Agent evaluation

• Production deployment

• Observability

• Security and governance

• Human-in-the-loop controls Reference Architecture USER │ ▼ Agent Gateway │ ▼ Orchestrator │ ┌──────────┼──────────┐ ↓ ↓ ↓ Research Data Agent Action Agent Agent │ │ │ │ └──────┼──────┘ ↓ MCP / A2A ↓ ┌──────┼──────┐ ↓ ↓ ↓ Enterprise RAG / APIs / Data Retrieval Tools │ │ │ └──────┼──────┘ ↓ Agent Runtime ↓ Evaluation + Monitoring ↓ Security + Governance

Certification details

Upon successful completion of the training, participants will be prepared with the knowledge and practical skills aligned to the Professional Agentic Architect certification exam.

CloudThat Course Completion Certificate will be awarded to learners who successfully complete the training.

The official exam guide describes the Professional Agentic Architect as a practitioner who designs and manages autonomous AI-driven workflows while considering reliability, performance, cost, security, and scalability.

FAQs

What is this course about?

This is an advanced course focused on designing, developing, evaluating, deploying, securing, and governing enterprise agentic AI solutions.

Is this a generic Generative AI course?

No. The focus is specifically on agent architecture and agentic application development, including ADK, MCP, A2A, RAG, agent memory, evaluation, orchestration, security, and governance.

Does the course include hands-on agent development?

Yes. Learners build custom agents and multi-agent workflows using ADK and related agent technologies.

Does the course cover MCP?

Yes. MCP is covered both for coding-agent integrations and custom agent/tool integrations.

Does the course cover multi-agent systems?

Yes. Sequential, parallel, graph-based workflows, agent handoffs, MCP, and A2A are covered.

Does the course cover RAG?

Yes. Enterprise RAG, embeddings, vector retrieval, similarity scoring, reranking, Agent Retrieval, Vector Search, and RAG Engine are covered.

Does the course cover production deployment?

Yes. Agent Runtime, Cloud Run, GKE, scaling, monitoring, troubleshooting, and optimization are included.

Does the course cover security?

Yes. Agent Identity, OAuth 2.0, Agent Gateway, Model Armor, access boundaries, guardrails, and human-in-the-loop controls are covered.

Is evaluation covered?

Yes. Learners create test sets, golden datasets, evaluation workflows, and continuous evaluation pipelines.

Is there a capstone project?

Yes. Learners design an enterprise multi-agent architecture incorporating development, RAG, orchestration, evaluation, deployment, observability, and security.