

AI AGENTS WITH GOOGLE'S ADK BOOTCAMP: BUILD WITH GEMINI, PROTOCOLS & AGENTOPS

DURATION

Format Instructor-Led Training (ILT or Instructed-Led Online (ILO))

Duration 4 days

PDU

This course will contribute 8 PMI® professional development units (PDUs) towards your chosen certification.

INTENDED FOR

- Software engineers or application developers familiar with Python who want a structured introduction to incorporating AI agent capabilities into applications
- AI/ML engineers who want to understand the fundamentals of a dedicated agent development framework like ADK
- Technical leads or architects who need a concise overview of AI agent development capabilities to inform technical strategy, without requiring deep hands-on implementation during the sessions
- Technically proficient professionals (Python, CLI) curious about AI agents, seeking a time-efficient, guided learning path to grasp the core concepts of ADK

Explore AI agents with Google's Agent Development Kit. You'll learn the core concepts, tools, and patterns you need to start creating agents that can reason, use tools, and interact with the world—all within Vertex AI. You'll be guided through ADK setup, run your first agents, integrate powerful language models like Gemini, add custom tools, manage basic conversational memory, and understand the important agentic patterns. You'll also dive into extending agents and making them interoperable with the Model Context Protocol and Agent-to-Agent protocol. Concluding the course are essential concepts around evaluation, deployment strategies, and crucial security practices. Each day is structured as a 2.5-hour session (two 75-minute blocks with a break in between), blending presentations, live demonstrations, hands-on exercises, and group discussions.

LEARNING OUTCOMES

- Understand the core concepts and workflow of building AI agents with ADK
- Run and test agents using ADK command-line tools (adk init, adk run, adk web, adk eval)
- Connect agents to Google's foundational LLMs (Gemini 2.5 / 3) and other open models via LiteLLM
- Define and register custom tools using Python functions with proper schemas
- Implement conversational memory and understand persistence strategies
- Grasp the fundamentals of multi-agent systems and agentic design patterns
- Adopt agent protocols and standards with MCP and A2A

AI AGENTS WITH GOOGLE'S ADK BOOTCAMP: BUILD WITH GEMINI, PROTOCOLS & AGENTOPS

PREREQUISITES

- A computer with Python installed
- A code editor (VS Code recommended)
- Internet access
- A Google Cloud Platform account with billing enabled (free credits available)
- Basic proficiency in Python programming
- Familiarity with fundamental cloud computing concepts (APIs, services)
- Understanding of core LLM concepts (prompting, RAG)
- Comfort using the command line / terminal

CONTENT

Day 1

Session 1: Introduction to Agents and ADK

- **Presentation:** What are AI agents and why are they relevant now?; overview of Agent Development Kit; core concepts (agent, tool, memory, workflow); the AI agent landscape (ADK vs. LangChain vs. Crew AI vs. LangGraph)
- **Demonstration:** ADK project structure; adk init; adk web for interactive testing
- **Hands-on exercise:** Run adk init to create a sample agent structure and familiarize yourself with the generated files
- **Group discussion:** What's one simple, repetitive task you think an AI agent could automate?
- **Q & A**

Session 2: Models and Tools

- **Presentation:** How agents use LLMs; connecting ADK to Vertex AI Gemini (config, code examples); Gemini model selection (2.5 Flash, 2.5 Pro, Gemini 3 Flash preview); connecting ADK to open-source models with LiteLLM; defining basic tools (Python functions); tool schemas and parameter types
- **Demonstration:** Agent calling Gemini; agent calling a simple local tool with schema validation
- **Group discussion:** Beyond calculations, what kind of external information access would be useful as a tool for an agent?
- **Hands-on exercises:** Configure agent to connect to Vertex AI Gemini; implement a simple Python function (e.g., return current date/time) and register it as a tool in your agent
- **Q&A**

Day 2

Session 3: Multimodal Agents and Memory Persistence

- **Presentation:** What is multimodality?; conceptual flow—using Gemini's native multimodal capabilities via ADK; persistence and the role of memory; memory types overview (buffer, summary); implementing memory with ADK; artifacts and state management
- **Demonstration:** Agent using buffer memory to recall previous turns; agent processing an image input
- **Group discussion:** If an agent processes an image, what kind of memory or artifact might it need to store about it?

AI AGENTS WITH GOOGLE'S ADK BOOTCAMP: BUILD WITH GEMINI, PROTOCOLS & AGENTOPS

- **Hands-on exercises:** Add memory to your agent; use adk web to interact multiple times and observe how context influences responses
- **Q&A**

Session 4: Agentic Design Patterns

- **Presentation:** Why use design patterns?; retrieval-augmented generation (RAG)—implementing RAG conceptually with ADK (using a search/lookup tool and LLM); sub-agent patterns (sequential agents, delegate agents, loop agents); implementing other agentic patterns; when to use multi-agent vs. single-agent architectures
- **Demonstration:** Agentic design patterns in action; sub-agent delegation flow
- **Group discussion:** What pattern would be interesting to explore to implement your use case?
- **Hands-on exercise:** Choose one of the discussed patterns and implement it
- **Q&A**

Day 3

Session 5: Putting It All Together

- **Presentation:** Revisiting the fundamental building blocks of an intelligent agent; how ADK simplifies the integration of these components; the ADK agent lifecycle from development to deployment
- **Group discussion:** What was your biggest “aha!” moment from Day 1 and Day 2?; Were there any concepts or exercises that you found particularly challenging?
- **Hands-on exercise:** The daily briefing agent—build a multi-tool agent that combines concepts from Days 1–2

Session 6: MCP and A2A

- **Presentation:** Model Context Protocol (MCP): standard for connecting agents to external tools and services; calling an MCP server from your ADK agent; Agent-to-Agent (A2A) communication: multi-agent collaboration protocol purpose and standards; high-level use cases; making your agent A2A-compatible
- **Demonstration:** Enabling your agents with MCP and A2A; swapping a custom tool for an MCP server tool
- **Group discussion:** What might be distinct roles for different agents in an e-commerce system (e.g., search agent, recommendation agent, checkout agent)?
- **Hands-on exercises:** Swap one of your agent tools with an existing tool from an MCP server; make your agent A2A-compatible
- **Q&A**

Day 4

Session 7: Agent Evaluation, Deployment, and AgentOps

- **Presentation:** Challenges in agent evaluation; building evaluation datasets; adk eval for running evaluations; AgentOps lifecycle introduction: key aspects (logging, tracing, monitoring, testing); deployment options: Vertex AI Agent Engine Runtime, Docker, Agent Starter Pack for CI/CD
- **Demonstration:** Agent eval run with a golden dataset; deployment to Agent Engine; Agent Starter Pack as an example of CI/CD
- **Group discussion:** What's one key difference between evaluating a traditional software application and evaluating an LLM-based agent?

AI AGENTS WITH GOOGLE'S ADK BOOTCAMP: BUILD WITH GEMINI, PROTOCOLS & AGENTOPS

- **Hands-on exercises:** Examine the detailed logs produced when running your agent with `adk run -v`; identify inputs, LLM calls, tool usage, and outputs; conceptually map how you might package your agent for deployment
- **Q&A**

Session 8: Agentic Security

- **Presentation:** Why agent security is critical: agents have tool access and data access; key risks (prompt injection, insecure tool handling, data privacy/leakage, overreliance); overview of mitigation strategies in ADK (input sanitization, tool permissions, rate limiting, human review); Model Armor: Google's guardrail framework for production agents; Agent Engine Threat Detection for runtime monitoring; responsible AI considerations for agents
- **Group discussion:** How might you prevent a tool used by an agent from performing unintended harmful actions?
- **Hands-on exercises:** Review the OWASP Top 10 for Large Language Model Applications; identify one or two risks that apply directly to the agent you've built; explore the different capabilities in ADK to secure your agent

Wrap-Up and Q&A

- Key takeaways and recap of the 4-day journey
- Resources: ADK documentation, community repos, Agent Starter Pack
- Next steps for continued learning
- Final Q&A

Lecturing is kept to the minimum necessary, and most of the learning is achieved by applying the practices and techniques in group exercises. Our LiveOnline delivery features an instructor who is 100% live, and interaction and learning objectives are the same as our in-person classes, with the added benefit of taking this course from your home or office. Since this class is delivered over half-days, it allows for greater flexibility and time for other work or activities.