

AI-500T00: Design and Implement Multi-Agent AI Solutions

Course ID #: 7000-1223-ZZ-Z

Hours: 28

Delivery Method: Group Internet Based

Course Content

Description:

In this course, you will cover: This course focuses on the practical skills needed to architect and develop multi-agent AI solutions using Microsoft Foundry and Azure, validating your ability to design logical architecture for multi-agent solutions, build and integrate tool ecosystems, implement multi-agent orchestration and integration of monitoring, security and governance.

Objectives:

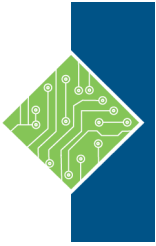
Upon successful completion of this course, students will:

- Design stateful agentic loops, session state, reflection, planning, and conversation-resumption patterns.
- Implement advanced multi-agent orchestration, task decomposition, handoff, and A2A communication architectures.
- Design production prompting, guardrail, MCP tool, advanced RAG, and durable agent-memory capabilities.
- Implement coordinated CI/CD pipelines and progressive deployment strategies for multi-agent systems.
- Apply zero-trust identity, secrets management, tenant isolation, encryption, and compliance controls.
- Establish responsible AI governance and enterprise agent lifecycle management practices.
- Implement distributed observability, structured logging, dashboards, and anomaly detection.
- Design multi-agent evaluation frameworks, synthetic test datasets, and behavioral regression testing.
- Optimize model routing, caching, token usage, context management, cost, quality, and latency.
- Design human-in-the-loop approval, feedback, and audit workflows.
- Debug production incidents using agent replay, root cause analysis, automated remediation, and post-mortem processes.

Prerequisites:

Suggested prerequisites:

- Completion of AI-103, Develop AI agents on Azure, and AI-300, Operationalize generative AI applications, or equivalent hands-on experience.
- Working experience with Microsoft Foundry Agent Service and Microsoft Agent Framework.
- Familiarity with multi-agent orchestration, MCP and custom function tools, RAG with Azure AI Search, OpenTelemetry, Azure Monitor, GitHub Actions, Azure managed identities and security services, responsible AI concepts, and Power Automate.
- Proficiency in Python, asynchronous programming, REST APIs, and the Azure AI and Azure OpenAI SDKs.



AI-500T00: Design and Implement Multi-Agent AI Solutions

Course ID #: 7000-1223-ZZ-Z

Hours: 28

Delivery Method: Group Internet Based

Target Audience:

- AI Engineers
- Solution Architects
- Developers and AI practitioners who design, build, secure, govern, monitor, and operate production-ready multi-agent AI solutions

Topics:

Topic 1: Architect Production-Grade Multi-Agent AI Solutions in Azure

Lesson 1: Design Stateful Agentic Loops with Microsoft Foundry Agent Service

- Examine production agentic loop architecture
- Examine the Foundry Responses API and Agents v2 model
- Implement agent reflection and planning cycles
- Design session state and context management
- Implement fork-based sessions and conversation resumption
- Migrate stateful agentic loops from Agents v1 to Agents v2

Lesson 2: Implement Advanced Multi-Agent Orchestration Patterns in Microsoft Foundry

- Differentiate agentic AI from multi-agent AI architectures
- Examine advanced orchestration architectures
- Implement hub-and-spoke orchestration
- Design parallel agent spawning and synchronization
- Compare orchestration frameworks

Lesson 3: Apply Task Decomposition and Agent Collaboration Strategies in Microsoft Foundry

- Design prompt chaining workflows
- Implement dynamic adaptive task decomposition
- Design agent handoff message schemas
- Ensure handoff reliability and context preservation
- Optimize decomposition granularity

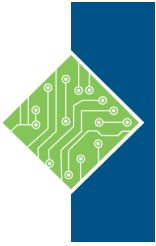
Lesson 4: Design Enterprise-Scale Agent Communication with A2A in Azure

- Design A2A agent ecosystems at scale
- Implement distributed shared state management
- Design context isolation and sharing strategies
- Build conflict detection and resolution mechanisms
- Resolve conflicts and maintain audit trails

Topic 2: Build Production-Grade Multi-Agent Capabilities with Microsoft Foundry

Lesson 5: Design Advanced Prompting Strategies for Production AI Agents

- Design multiturn reasoning prompt architectures
- Implement prompt injection defenses
- Build system prompt frameworks for agent control
- Design multi-intervention guardrail



AI-500T00: Design and Implement Multi-Agent AI Solutions

Course ID #: 7000-1223-ZZ-Z

Hours: 28

Delivery Method: Group Internet Based

architectures

- Implement prompt versioning and optimization
- Automate prompt regression and optimization
- Design fine-tuning strategy and data pipelines

Lesson 6: Build Enterprise-Grade Tool Ecosystems with MCP and Microsoft Foundry

- Design production MCP server architecture
- Build MCP servers with error handling and fallback
- Implement tool selection and routing logic
- Govern tool dependencies and versioning

Lesson 7: Implement Advanced RAG Pipelines with Azure AI Search and Microsoft Foundry

- Design hybrid search architectures
- Implement reranking and context ranking
- Design dynamic knowledge source routing
- Optimize chunking and embedding strategies

Lesson 8: Design Multi-Agent Memory Architectures with Azure Cosmos DB

- Examine memory architecture patterns
- Implement semantic memory with vector storage
- Optimize memory retrieval and context injection
- Configure context window optimization
- Design memory retention and consolidation
- Enforce memory privacy and audit compliance

Topic 3: Deploy and Govern Enterprise Agentic AI Solutions on Azure

Lesson 9: Implement CI/CD Pipelines for Multi-Agent Systems with GitHub Actions

- Design multi-agent deployment pipelines
- Implement progressive deployment strategies
- Configure multi-environment agent deployment strategies
- Automate rollback procedures

Lesson 10: Secure Multi-Agent Systems with Azure Zero-Trust Architecture

- Apply zero-trust identity to agent networks
- Secure agent access with JIT and workload identity
- Design authentication flows and secrets lifecycle
- Prevent lateral movement in agent networks
- Implement tenant context propagation and data isolation
- Validate tenant boundaries and enforce encryption
- Configure compliance controls for regulated

agent deployments

Lesson 11: Scale Responsible AI Governance with Azure AI Content Safety and Microsoft Foundry

- Design fairness and bias monitoring
- Implement transparency and explainability
- Configure privacy protection in multi-agent workflows
- Establish audit and accountability frameworks

Lesson 12: Govern the Enterprise Agent Lifecycle in Microsoft Foundry

- Design agent versioning and approval workflows
- Implement usage quotas and rate limiting
- Design cost allocation and chargeback models
- Establish agent retirement and deprecation processes



AI-500T00: Design and Implement Multi-Agent AI Solutions

Course ID #: 7000-1223-ZZ-Z

Hours: 28

Delivery Method: Group Internet Based

Topic 4: Monitor, Evaluate, and Operate Multi-Agent AI Solutions in Azure

Lesson 13: Implement Distributed Observability for Multi-Agent Solutions with OpenTelemetry

- Design distributed tracing for multi-agent solutions
- Implement structured logging for agent decisions
- Configure telemetry aggregation and dashboards
- Build anomaly detection for agent behavior

Lesson 14: Design Evaluation Frameworks for Multi-Agent Solutions with Microsoft Foundry

- Define multi-agent success metrics
- Implement LLM-as-judge evaluation for multi-agent systems
- Design synthetic test datasets for multi-agent evaluation
- Build regression testing pipelines to detect agent drift

Lesson 15: Optimize Multi-Agent Performance and Cost in Microsoft Foundry

- Design model routing for agent ecosystems
- Implement multi-level caching strategies

- Optimize token usage and context management
- Balance quality, cost, and latency tradeoffs

Lesson 16: Design Human-in-the-Loop Approval Workflows with Power Automate and Microsoft Teams

- Design confidence-based escalation for human intervention
- Implement approval workflows for agent-initiated actions
- Build active learning from human feedback
- Configure audit workflows for regulated decisions

Lesson 17: Debug and Respond to Production Multi-Agent Incidents in Azure

- Implement agent replay for production debugging
- Design root cause analysis for agent failures
- Configure automated incident detection and remediation
- Establish incident response and post-mortem processes

Register for this class by visiting us at:

www.tcworkshop.com or by calling us at 800-639-3535

NASBA CPE details are provided on the following pages.



AI-500T00: Design and Implement Multi-Agent AI Solutions

Course ID #: 7000-1223-ZZ-Z

Hours: 28

Delivery Method: Group Internet Based

NASBA Information

Level: Advanced

Advanced Preparation: See suggested prerequisites above.

Attendance Requirement: To be awarded the full credit hours, you must sign in and attend the entire course.

Recommended Field(s) of Study: Computer Software & Applications

Recommended CPEs: 31.20

Policies: Course Registration, Cancellation, Refund, and Complaint Resolution

For more information regarding administrative policies such as complaint and program cancellation policies, please contact our offices at 800-639-3535 or visit us at: www.tcworkshop.com

Official National Registry Statement:

The Computer Workshop is registered with the National Association of State Boards of Accountancy (NASBA) as a sponsor of continuing professional education on the National Registry of CPE Sponsors. State boards of accountancy have final authority on the acceptance of individual courses for CPE credits. Complaints regarding registered sponsors may be submitted to the National Registry of CPE Sponsors through its website: www.nasbaregistry.org

NOTE: Since our information is in multiple places on our website or in PDF format that is sent to clients, we have provided our normal course content with the NASBA Information added along with links to our policy page on the web. We will add our name to the Official National Registry Statement after we are approved.