Mastering Microsoft Copilot Studio

Duration: 3 Days Course Code: M55628A Delivery Method: Company Event

Overview:

This course is an immersive, hands-on training course designed to equip IT professionals, solution architects, and business technologists with the skills to design, build, and manage intelligent AI agents using Microsoft Copilot Studio.
Participants will gain practical experience creating both conversational and autonomous agents that integrate seamlessly with Microsoft 365, Power Platform, and enterprise data sources. Through guided labs, real-world examples, and advanced customization exercises, learners will develop the confidence to deploy secure, scalable Copilot agents that enhance productivity, automate workflows, and transform organizational processes.
This course blends conceptual understanding with hands-on application, progressing from foundational Copilot principles to advanced orchestration, governance, and autonomous agent design.

Updated July 2026

Company Events

These events can be delivered exclusively for your company at our locations or yours, specifically for your delegates and your needs. The Company Events can be tailored or standard course deliveries.

Target Audience:

This course is designed for professionals who want to build or manage intelligent agents using Microsoft Copilot Studio, including:
- Power Platform Makers and Developers who wish to extend their automation capabilities with AI-driven agents.
- IT Administrators and Solution Architects responsible for deploying and governing AI solutions within Microsoft 365 environments.
- Business Analysts and Citizen Developers looking to create task-specific copilots without extensive coding.
- AI and Automation Specialists who want to design secure, autonomous, and scalable Copilot agents for enterprise use cases.

Objectives:

- **After this course participants should be able to:**
- Create and configure copilots using Microsoft Copilot Studio.
- Design conversational experiences using topics, entities, and generative AI capabilities.
- Integrate copilots with Microsoft 365, Power Platform, and external data sources.
- Implement actions and workflows to automate business processes.
- Configure authentication, security, and user access settings for copilots.
- Test, publish, and manage copilots for organizational use.
- Monitor copilot performance and optimize responses to improve user experience.

Prerequisites:

Participants are recommended to have a basic understanding of Microsoft 365, Power Platform concepts, and business process automation, along with familiarity with generative AI and conversational AI solutions.

Testing and Certification

Follow-on-Courses:

None

Content:

Module 1 – Copilot Chat and Agents

Introduce participants to Microsoft Copilot Chat and the concept of agents within the Microsoft 365 ecosystem. Learners will explore how Copilot serves as the AI interface for users and how agents extend its capabilities to perform tasks, retrieve data, and automate workflows.

- Understanding Microsoft Copilot
- Overview of Copilot in Microsoft 365 (Teams, Word, Outlook, and more)
- How Copilot Chat enhances productivity through conversational AI
- “Copilot lite” vs “Full Copilot” capabilities
- How Copilot integrates with organizational data and permissions
- Introducing Agents
- Definition and purpose of Copilot Agents
- Types of agents: Simple, Task-based, and Autonomous
- Example scenarios: IT Helpdesk Bot, HR Policy Assistant, Project Tracker
- How agents interact with Copilot Chat
- Deployment and Security
- Where agents run: Teams, web, and Power Platform environments
- Data governance, privacy, and compliance considerations
- Licensing, cost control, and telemetry tracking
- Lab 1: Build agents in Copilot Studio Lite Experience

Module 2 – Building Agents with Copilot Studio

This module provides a deep dive into the architecture, core capabilities, and authoring experience of Microsoft Copilot Studio. Participants will learn how Copilot Studio operates within the Microsoft ecosystem, understand its performance and governance considerations, and develop hands-on skills in building conversational agents using the authoring canvas and its core components followed by an introduction to its advanced tools, orchestration features, and generative AI integration.

Through guided instruction and a practical lab, learners will design, author, and deploy a fully functional conversational agent using variables, entities, and structured conversation flows.

- Copilot Architecture and Core Capabilities
- Overview of Copilot Architecture and its role in the Microsoft 365 and Power Platform ecosystem
- Copilot Studio Core Capabilities – building, deploying, and managing conversational and

This module introduces learners to the power of Retrieval-Augmented Generation (RAG) and how it enhances Copilot Studio agents with accurate, grounded, and secure generative responses.

Participants will explore how RAG architecture operates within Copilot Studio, how to connect and manage knowledge sources effectively, and how to safely infuse generative AI capabilities

into topic designs while maintaining compliance and reliability.

- How RAG Enhances AI Responses
- Understanding Retrieval-Augmented Generation (RAG) and why it’s essential for enterprise-grade AI
- The limitations of pure generative AI (hallucination, context drift, factual inaccuracy)
- How RAG combines retrieved data from trusted knowledge sources with generative reasoning
- Benefits of RAG in Copilot Studio:
- Grounded responses based on organizational data
- Context retention and relevance in extended conversations
- Reduced risk of misinformation or unsupported answers
- Real-world examples: HR policy retrieval, compliance support, and project insights
- RAG Architecture in Copilot Studio
- Overview of RAG architecture within the Microsoft Copilot ecosystem
- The retrieval layer: indexing, search, and grounding data pipelines
- The generation layer: contextual synthesis using large language models (LLMs)
- How Copilot Studio manages context windows, relevance scoring, and ranking
- Integrating with Microsoft Graph, SharePoint, Dataverse, and external data repositories
- Understanding caching, latency, and query optimization in RAG-based designs
- Knowledge Sources and Generative AI
- Defining knowledge sources in Copilot Studio
- SharePoint document libraries
- Dataverse tables
- External data via connectors or APIs
- How Copilot Studio uses knowledge sources to ground generative responses
- Techniques for data curation and preparation for RAG indexing
- Integrating structured and unstructured data into your Copilot agent

Understanding ALM Strategy

- Defining an ALM strategy for Copilot agents within Microsoft 365 and Power Platform
- Why ALM is essential for enterprise-scale development and governance
- The benefits of structured lifecycle management:
- Version control
- Testing and quality assurance
- Controlled deployment across environments
- Reduced risk and rework
- Aligning ALM practices with organizational change management and security policies
- What Is ALM and Why It’s Important
- Overview of Application Lifecycle Management (ALM) concepts
- Development ? Testing ? Staging ? Production cycles
- Managing agent versions and configurations
- How ALM supports collaboration between makers, developers, and administrators
- Common pitfalls in unmanaged agent updates or direct publishing
- Real-world ALM examples in Copilot Studio agent deployment
- What “Publish” Really Does in Copilot Studio
- Understanding the Publish process in Copilot Studio
- What happens behind the scenes when publishing an agent
- Version creation, environment packaging, and synchronization
- How publishing differs from exporting/importing solutions in Power Platform
- Best practices for publishing safely without disrupting production agents
- Integrating publishing into your broader ALM workflow
- Power Platform ALM for Copilot Studio
- Overview of Power Platform ALM capabilities relevant to Copilot Studio
- How solutions encapsulate Copilot agents, connections, and data configurations
- Managing Copilot components as part of broader Power Platform solutions
- Understanding environments and their role in ALM:
- Development, Test, UAT, and Production
- Environment permissions, data boundaries, and DLP alignment
- Tracking agent versions, dependencies, and solution history
- ALM with Azure DevOps
- Integrating Azure DevOps with Power Platform and Copilot Studio
- Managing Copilot solution source control and version tracking

autonomous agents
- Copilots and Conversational AI – the evolution from chat-based interaction to orchestrated reasoning
- Understanding conversation volumes, quotas, and limits for performance management
- Performance optimization and telemetry insights for efficient operations
- Planning and Lifecycle Management
- Planning your agent – defining purpose, scope, target users, and measurable outcomes
- The Typical Copilot Studio Lifecycle:
- Initiate ? Prepare ? Design ? Build ? Deploy ? Operate ? Optimize
- Selecting the right tools for your governance requirements
- Citizen, Partnered, and Pro Developer governance zones
- Role-based permissions and security boundaries
- Leveraging the Power Platform Center of Excellence (CoE) for governance, auditing, and analytics
- Copilot Studio Authoring Canvas
- Navigating the authoring canvas and understanding topic-driven design
- Using the Message Node to deliver static, dynamic, or formatted responses
- Using the Question Node to capture user input and enable conditional branching
- Creating rich text responses (hyperlinks, adaptive cards, lists, media)
- Using variables to navigate customers to tailored content
- Defining entities and slot-filling for structured data collection
- Topic management – organizing and linking topics for efficient conversation flow
- Using enhanced speech authoring capabilities for voice interactions
- Productivity and pro-code options – integrating Power Fx, expressions, and custom logic
- Lab 2: Build a conversational agent in Copilot Studio
- Agent Tools, Knowledge, and Orchestration
- Overview of Agent Tools within Copilot Studio
- Connecting Knowledge Sources for grounding answers and context
- Using Tools to extend agent capabilities:
- Copilot Connectors – integrate with Microsoft 365, Dynamics 365, and external APIs
- Computer Use Tool – automate interactions with desktop or web apps
- Code Interpreter – execute Python code directly within Copilot Studio
- Advanced Approvals – automate multi-level approval processes
- Leveraging the Orchestrator to coordinate multi-topic, multi-tool execution
- Designing Prompts effectively for contextual,
- How Generative AI interacts with these sources to produce reliable, factual answers
- Generative AI Security and Compliance Considerations
- Data protection in generative AI workflows
- How Copilot Studio handles sensitive or restricted content
- Managing compliance requirements (GDPR, data residency, retention policies)
- Understanding Responsible AI principles within Microsoft's Copilot framework
- Limiting generative scope through instructions, topic constraints, and data access rules
- Setting boundaries for autonomous or open-ended responses
- Infusing Generative AI into Topics
- Methods for embedding generative capabilities directly within topics
- Enabling Generative Building and Generative Answers in Copilot Studio
- Designing hybrid topics that use both structured and generative logic
- Writing grounded prompts that combine user input, retrieved knowledge, and AI synthesis
- Testing and refining generative topics to ensure consistency, tone, and accuracy

Module 6 – Integrations

This module explores how to integrate Copilot Studio agents with enterprise systems and external data sources using connectors, flows, APIs, and automation tools.

Learners will understand key integration patterns, performance constraints, and quotas, as well as how to extend Copilot capabilities through custom connectors, HTTP requests, and the

Model Context Protocol (MCP).

- Integration Patterns and Considerations
- Overview of integration architecture within Copilot Studio and the Power Platform
- Integration patterns:
- Direct integration (connectors and HTTP actions)
- Event-driven orchestration (Power Automate flows)
- Data-driven integration (Dataverse, APIs, Azure Logic Apps)
- Choosing the right integration model based on scalability, latency, and control
- Security considerations for data flow and authentication (OAuth, managed identity,
- Automating deployment pipelines through Azure DevOps YAML templates
- Example workflows:
- Export ? Validate ? Deploy
- Trigger-based deployments for agents or connectors
- Using Azure DevOps Boards for ALM governance and change tracking
- GitHub Actions for Microsoft Power Platform
- Introduction to GitHub Actions for CI/CD with Power Platform
- Setting up a GitHub repository to manage Copilot Studio solutions
- Example automation:
- Exporting a solution from Dev ? Importing to Test or Prod
- Running validation checks before deployment
- Comparing Azure DevOps Pipelines vs. GitHub Actions for ALM workflows
- Security and permission considerations for GitHub integrations
- Power Platform Pipelines for Copilot Studio
- Introduction to Power Platform Pipelines — no-code ALM for citizen and pro developers
- How Pipelines simplify solution promotion across environments
- Configuring automated pipelines for Copilot Studio agents
- Using deployment profiles to control environment variables and data connections
- Monitoring deployment success, rollback procedures, and version tracking
- Combining Pipelines, GitHub, and DevOps for hybrid ALM strategies

Module 9 – Analytics ; KPIs

This module teaches learners how to measure, analyze, and optimize the performance of Copilot Studio agents using data-driven insights.

Participants will explore conversation analytics, engagement metrics, and key performance indicators (KPIs) that reflect business impact and user satisfaction.

By implementing a structured analytics and optimization strategy, learners will be able to continuously improve their agents' effectiveness, refine conversation design, and demonstrate ROI to stakeholders.

- Conversation Design and Outcome Tracking
- Understanding conversation analytics in Copilot Studio

human-like responses
- Generative AI and Optimization
- Generative AI in Copilot Studio – enabling AI reasoning and dynamic responses
- Generative building – AI-assisted topic creation and optimization
- Generative answers – grounded, data-informed AI responses
- How Retrieval-Augmented Generation (RAG) supports accuracy and reliability
- Using analytics to easily optimize with data-driven insights
- Tracking agent performance
- Measuring engagement and response accuracy
- Iterating designs based on telemetry
- Integration, Agent Flows, and Multi-Agent Orchestration
- Designing Agent Flows for end-to-end automation
- Multi-agent orchestration – coordinating several specialized agents to handle complex workflows
- Planning integrations with Power Automate, Azure services, and external APIs
- Deploying and using agents in any system – Teams, web, Dynamics 365, or embedded applications
- Lab 3: Use tools in Copilot Studio

Module 3 – Building Autonomous Agents

This module introduces the concept of autonomous agents within Microsoft Copilot Studio and explores how they extend beyond conversational interactions to perform independent, goal-oriented actions.

Learners will understand the difference between conversational and autonomous agents, explore the core components that enable autonomy, and learn how to craft precise, ethical, and

effective agent instructions.

Through guided demonstrations and practical exercises, participants will design the foundational structure of an autonomous agent, integrating triggers, tools, and logic to automate business processes intelligently and responsibly.

- Understanding Autonomous Agents
- What Are Autonomous Agents?
- Definition and evolution of autonomous agents within Microsoft Copilot Studio
- How autonomous behavior enhances business automation and decision-making
- Examples of autonomous agents in real-world enterprise scenarios (approvals, service principal)
- Quotas, Limits, and Performance
- Understanding Copilot Studio integration quotas and limits (calls per minute, session size, data throughput)
- Performance tuning strategies for efficient agent workflows
- Managing API throttling, retries, and error handling in long-running tasks
- Using telemetry and analytics to monitor connector and flow performance
- How to design integrations that minimize cost and resource consumption
- Agent Flows, HTTP Actions, and Connectors
- Introduction to Agent Flows in Copilot Studio for orchestrating integrations
- Using HTTP actions for RESTful API calls and external service integration
- Working with standard connectors (SharePoint, Outlook, Teams, Azure, Dynamics 365, Dataverse, etc.)
- Designing connector-based actions to extend agent functionality
- Handling timeouts and long-running processes in cloud flows
- Best practices for async processing and status callbacks
- Managing cloud flow timeouts gracefully with user feedback loops
- Custom Connectors and Advanced Integration
- Overview of Custom Connectors in Copilot Studio and Power Platform
- How to build and register a custom connector for internal or external APIs
- Using Swagger/OpenAPI definitions to define connector actions and responses
- Testing and validating custom connectors in sandbox environments
- Governance considerations for connector publishing and sharing
- Integration with Azure Functions and Logic Apps for extensibility
- Model Context Protocol (MCP)
- Introduction to the Model Context Protocol (MCP) and its role in AI-driven integrations
- How MCP enables Copilot agents to securely access external data models
- Using MCP to extend Copilot with contextual understanding from multiple systems
- Best practices for maintaining data integrity and compliance in MCP-connected agents
- Computer-Using Agents (CUA) and RPA
- Understanding Computer-Using Agents (CUA) – what they are and how they work
- Enabling agents to interact with desktop applications and web browsers
- Comparing CUA and Robotic Process Automation (RPA):
- RPA for structured, rule-based workflows
- CUA for intelligent, adaptive automation
- Measuring conversation flow effectiveness: intent recognition, success paths, and drop-off points
- Designing conversations with measurable outcomes (e.g., task completion, satisfaction, resolution rates)
- Tracking end-user interactions and intent success using telemetry and built-in analytics
- Mapping conversational outcomes to business objectives and performance goals
- Using conversation data to refine prompts, topics, and agent logic
- Engagement and Outcomes
- Defining and measuring engagement metrics:
- Total users, active sessions, conversation depth, and dwell time
- Repeat interactions and user satisfaction trends
- Identifying key engagement drivers — tone, context, personalization, and response time
- Using data to segment audiences and identify high-value user scenarios
- Correlating agent engagement with organizational productivity and ROI
- Example KPIs:
- Resolution rate per topic
- Time-to-response improvement
- Reduction in support tickets through automation
- Business cost savings from AI adoption
- Analytics Strategy
- Building a comprehensive analytics strategy for Copilot agents
- Aligning analytics goals with business priorities and governance policies
- Defining measurable KPIs for agent performance and value realization
- Using Microsoft analytics tools:
- Copilot Studio Analytics Dashboard
- Power BI integration for advanced reporting
- Dataverse telemetry for raw data analysis
- How to combine Copilot analytics with Power Platform CoE dashboards
- Tracking metrics across environments: Dev, Test, and Production
- Data governance considerations in analytics — ensuring accuracy and privacy
- Optimization Strategy
- Developing an optimization cycle for continuous improvement:
- Monitor ? 2. Analyze ? 3. Adjust ? 4. Deploy ? 5. Measure again
- Leveraging A/B testing for topic or prompt improvements
- Applying analytics insights to refine:
- Agent tone and conversational flow
- Topic hierarchy and orchestration logic
- Knowledge source selection and generative AI prompts
- Using performance data to identify

notifications, data insights)
- Conversational vs Autonomous Agents
- Key differences in purpose, behavior, and user interaction
- Conversational agents: reactive and guided by prompts
- Autonomous agents: proactive, event-driven, and task-focused
- Choosing the right agent type for your business process
- Reinventing Business Processes with Agents
- How autonomous agents can reimagine workflows across departments
- Mapping manual processes to autonomous agent capabilities
- Best practices for balancing automation with control and accountability
- Crafting Autonomous Agent Instructions
- The Art of Instruction Design
- Writing clear, structured agent instructions
- Defining role, tone, scope, and context for autonomous behaviour
- Handling ambiguity, uncertainty, and ethical constraints
- Building Blocks of Writing Agent Instructions
- Purpose statement
- Behavioural rules and escalation paths
- Contextual memory and grounding
- Error handling and safety checks
- Examples of well-structured vs poorly written instructions
- Building Blocks of an Autonomous Agent
- Core Components of Autonomy
- Triggers: initiating events that activate agents (manual, scheduled, event-driven)
- Tools: actions and connectors that allow agents to perform work
- Knowledge: content sources and grounding data
- Instructions: defining how agents reason and respond
- Designing multi-step workflows with Copilot Studio tools and triggers
- Understanding dependencies, data flow, and governance boundaries
- Lab 4: Make your agent autonomous in Copilot Studio

Module 4 – Language ; Orchestration

This module focuses on how Microsoft Copilot Studio interprets, processes, and orchestrates natural language to deliver accurate, contextual, and human-like responses.

Learners will explore language understanding models, orchestration types, and best practices for building intelligent agents that can manage ambiguity, route topics effectively, and deliver consistent communication across languages and scenarios.

through AI orchestration
- Building integrated processes that combine cloud flows, connectors, and CUAs
- Example use cases:
- Reading data from legacy apps
- Filling forms or automating reports
- Coordinating actions between on-premises and cloud systems

Module 7 – Security, monitoring ; governance

This module equips learners with the knowledge and best practices needed to secure, monitor, and govern Microsoft Copilot Studio environments at scale.

Participants will learn how to balance innovation and control, implement zoned governance models, enforce data loss prevention (DLP) and compliance policies, and manage the security of both agents and users.

- Balancing Innovation and Governance
- The importance of governance in AI and Copilot deployment
- Balancing citizen development and enterprise oversight
- How governance enables safe innovation without stifling productivity
- Building governance frameworks aligned with corporate IT and security policies
- Common governance challenges in scaling Copilot adoption
- The Agent Controls Model
- Understanding the Agent Controls Model: policies, permissions, and oversight
- Key governance layers: user, environment, tenant, and data
- How agent-level controls support compliance and operational transparency
- Tracking agent performance and telemetry for security auditing
- Zoned Security, Governance, and Operations
- Overview of Governance Zones (1–3):
- Zone 1: Citizen Development (low-risk, innovation sandbox)
- Zone 2: Partnered Development (moderate risk, departmental use)
- Zone 3: Pro Development (high governance, enterprise-critical)
- Mapping organizational maturity to governance zones
- Getting Started with Zones – setting up secure, scalable environments
- How to manage transitions between zones while maintaining compliance
- Security and Administration Controls
- Overview of security architecture in Copilot Studio and Power Platform
- Security, agent, and user management

underperforming agents or topics
- Creating a feedback loop with stakeholders and users for ongoing tuning
- Setting thresholds and alerts for critical KPIs (e.g., low satisfaction, high failure rate

Module 10 – Licensing and capacity

This module provides learners with a deep understanding of how Microsoft Copilot Studio licensing and capacity consumption work within the Power Platform ecosystem.

Participants will learn how to plan, monitor, and manage Copilot Studio resource usage across environments while maintaining cost efficiency and operational scalability.

- Licensing and Capacity Overview
- Overview of Copilot Studio licensing models
- Licensing through Microsoft 365, Power Platform, and standalone Copilot subscriptions
- Key licensing dependencies: Power Virtual Agents, Power Automate, Dataverse
- Capacity components and what they represent:
- Dataverse storage (database, file, log)
- Power Platform request limits
- AI Builder and Copilot Studio usage entitlements
- Aligning licensing strategy with your organization’s scale, user base, and governance zones
- How to assign and manage licenses across tenants and environments
- Basic Credit Consumption Scenarios
- Understanding Copilot capacity credits and how they are consumed
- Common usage patterns that drive credit consumption:
- Agent interactions and conversation sessions
- Generative AI responses and knowledge retrieval
- Tool usage (code interpreter, connectors, RAG queries)
- Mapping agent types to credit requirements (simple, task-based, autonomous)
- Real-world credit usage examples for typical Copilot deployments
- How conversation complexity, orchestration, and integration affect credit burn
- Agent Activity and Billing Rates
- Understanding Agent Activity metrics and their impact on billing
- How billing rates differ based on:
- Generative AI vs. retrieval-based responses

- Natural Language Understanding (NLU) in Copilot Studio
- Overview of Natural Language Understanding and its role in conversational AI
- How Copilot interprets intent, entities, and context from user input
- Comparison of language processing models:
- Classic Orchestration (topic-based routing)
- Generative Orchestration (AI-driven reasoning and chaining)
- The evolution from rule-based intent matching to generative orchestration
- Orchestration Models and Logic
- Classic Orchestration
- How Copilot matches user intent to predefined topics
- Topic routing, priority, and fallback mechanisms
- Using disambiguation prompts when multiple intents are detected
- Disambiguation and Orchestration
- Strategies for managing ambiguous queries
- Designing topic structures for clear and efficient disambiguation
- Example: differentiating between “request access” vs. “reset password” topics
- Generative Orchestration
- How Copilot uses generative AI to route across tools, topics, and data
- The orchestration engine: connecting tools, knowledge sources, and actions
- Combining structured (classic) and generative (adaptive) orchestration for hybrid agents
- How orchestration improves responsiveness, reduces errors, and automates complex reasoning
- How Tools and Topics Are Orchestrated
- Role of the orchestrator in managing flow between topics and tools
- Understanding priority, context retention, and handoffs
- Integrating multiple data sources or services within a single agent session
- Designing for Clarity and Effectiveness
- Best Practices for Agent Instructions
- Writing clear, concise, and contextual instructions for accurate orchestration
- Controlling tone, scope, and fallback responses
- Using grounding and constraints to ensure reliable generative outputs
- Best Practices for Topic Inputs ; Outputs
- Defining input and output expectations for each topic
- Using metadata and variable binding for consistent data transfer
- Creating well-structured topic hierarchies to reduce confusion and error loops
- Language Control and Localization
- What Controls the Agent Language
- Role of AI models, user preferences, and system settings in language handling
- Managing multilingual environments with strategies:
- Assigning roles and permissions
- Enabling secure authentication (Azure AD, Entra ID)
- Monitoring user actions and agent activity logs
- Designing an effective environment strategy for production, test, and development
- Understanding Copilot Studio security roles and least-privilege access design
- Data Loss Prevention (DLP) and Policy Management
- Overview of DLP Policies in Power Platform and Copilot Studio
- The role of DLP connectors and data classification in controlling data flow
- How to manage connectors across risk profiles (Business vs. Non-Business)
- DLP policies and rules per environment — and when they can be safely relaxed
- Designing DLP frameworks that balance flexibility and compliance
- Practical examples: blocking external connectors, auditing data access
- Securing Copilot Studio Usage at Scale
- Strategies for secure scaling across large organizations
- Controlling adoption through environment boundaries and sharing rules
- Using analytics and reports to monitor agent performance, cost, and usage
- Automating governance checks and policy enforcement through CoE Starter Kit
- Prompt Injection Mitigations – preventing manipulation of agent behavior through malicious inputs
- DDoS Protection for Anonymous Chatbots – safeguarding public-facing Copilots against overload attacks
- Data Residency and Compliance Management
- Understanding data residency and storage in Microsoft Copilot Studio
- Managing data movement restrictions across geographies and tenants
- Compliance with GDPR, SOX, HIPAA, and other global standards
- Designing for multi-region governance and local data processing
- Tools and best practices for monitoring data movement and access patterns

Module 8 – Application Lifecycle Management

This module focuses on implementing Application Lifecycle Management (ALM) practices for Copilot Studio.

Learners will explore how ALM ensures structured development, testing, and deployment of Copilot agents across

- Tool and connector usage
- Frequency of orchestration or autonomous agent actions
- Reading and interpreting usage reports in the Power Platform admin center
- Using telemetry data to connect activity metrics to cost drivers
- Best practices for minimizing unnecessary agent calls and redundant executions
- Understanding Credit Burn Rate
- Definition of credit burn rate in Copilot Studio
- How to monitor and project credit consumption across environments
- Factors influencing burn rate:
- Agent concurrency and scaling
- Number of active users or sessions
- Size and complexity of AI prompts and retrieval operations
- Strategies for managing and reducing burn rate:
- Optimizing conversation length and efficiency
- Reusing knowledge sources and cached responses
- Scheduling non-critical agents during off-peak hours
- Setting up alerts or dashboards to track consumption trends
- Copilot Studio Estimator
- Introduction to the Copilot Studio Estimator Tool
- How to use the estimator to forecast usage, licensing, and costs
- Simulating scenarios based on:
- Number of agents
- Daily conversation volume
- Generative AI usage patterns
- Interpreting estimator outputs to guide budget and capacity planning
- Integrating estimator results into business case and ROI modeling
- Capacity Management
- Building a capacity management strategy for Copilot Studio
- Monitoring capacity in the Power Platform Admin Center
- Allocating capacity per environment and adjusting for usage growth
- Using governance zones and environment strategy to balance capacity
- Managing cross-tenant capacity sharing and reporting
- Planning for scale: forecasting enterprise-wide usage
- Coordinating with IT operations and finance teams for ongoing monitoring

Module 11 – Testing agents

This module teaches learners how to effectively test, validate, and ensure quality for Copilot Studio agents before deployment.

consistent tone and structure
- Auto-Detect Spoken Language
- How Copilot Studio detects and adapts to user language automatically
- Best practices for supporting multilingual users (voice and text)
- Configuring fallback or default language options for enterprise scenarios

Module 5 – AI capabilities

environments while maintaining governance, consistency, and control.

Participants will gain practical insights into using Power Platform ALM, Azure DevOps, GitHub Actions, and Power Platform Pipelines to manage agent updates and continuous delivery in enterprise environments.

Participants will explore Copilot Studio Kit testing capabilities, learn how to test agents at scale, and understand the types of tests supported within the Copilot Studio ecosystem.

By applying structured testing practices, learners will develop the skills to identify defects, improve performance, and deliver reliable, production-ready Copilot agents that align with enterprise standards.

- Introduction to Testing in Copilot Studio
- The importance of testing and validation in the Copilot agent lifecycle
- How testing fits into the Application Lifecycle Management (ALM) and deployment process
- Typical challenges in testing AI-driven and conversational systems
- Core goals of agent testing:
- Ensuring accuracy, reliability, and usability
- Verifying data access and security
- Confirming proper orchestration and workflow logic
- Overview of the Copilot Studio Kit
- Introduction to the Copilot Studio Kit and its testing capabilities
- Components of the Kit:
- Test automation framework
- Reporting and analytics tools
- Configuration and environment setup utilities
- How the Kit integrates with Power Platform, Azure DevOps, or GitHub Actions pipelines
- Preparing the testing environment and data sets
- Using the Kit for continuous testing in multi-environment deployments
- Testing Agents at Scale
- Strategies for scaling agent testing across multiple environments and use cases
- Simulating large-scale user interactions and load testing scenarios
- Managing and tracking test execution across Dev, UAT, and Production environments
- Automating tests as part of CI/CD pipelines (DevOps or GitHub Actions)
- Monitoring performance and response times under real-world usage conditions
- Identifying and resolving issues related to:
- Latency and orchestration delays
- Data retrieval and grounding (RAG) errors
- Tool integration failures or misconfigurations
- Ensuring governance compliance during automated test runs
- Supported Test Types in the Copilot Studio Kit
- Overview of test types supported in Copilot Studio Kit:

- Unit Tests: Validate individual topics, nodes, and responses
- Integration Tests: Verify that connectors, triggers, and tools function correctly together
- Regression Tests: Ensure existing functionality remains stable after changes or updates
- Performance Tests: Evaluate speed, concurrency, and response times
- Security Tests: Validate DLP adherence, authentication, and permissions
- Conversational Flow Tests: Assess natural language understanding, disambiguation, and orchestration accuracy
- Best practices for selecting appropriate test types for each phase of development
- How to interpret test results and generate actionable reports

Additional Labs:

The following labs have been provided in order to gain additional hands on learning experience with Copilot Studio. They augment the labs aligned to each module and should be carried out once all the previous labs have been completed if the course is run other 3 days:

- Advanced Lab 1: Create a Monthly Business Review (MBR) Agent
- Advanced Lab 2: Build an Autonomous Account News Assistant Agent
- Advanced Lab 3: Track conversation outcomes and user feedback on AI responses
- Advanced Lab 4: Autonomous Portfolio Lookup Agent with Computer-Using Agents (CUA)
- Advanced Lab 5: Deliver high-quality, scalable agents with Copilot Studio Kit
- Advanced Lab 6: Model Context Protocol (MCP) ; Copilot Studio

Further Information:

For More information, or to book your course, please call us on Head Office 01189 123456 / Northern Office 0113 242 5931

info@globalknowledge.co.uk

www.globalknowledge.com/en-gb/

Global Knowledge, Mulberry Business Park, Fishponds Road, Wokingham Berkshire RG41 2GY UK