AI CERTs®

AI+ Agent Specialty (1 Day)

Program Detailed Curriculum

Executive Summary

The AI+ Agent Specialty certification is designed to validate professionals' expertise in utilizing artificial intelligence tools and technologies to solve business problems. The certification covers key areas such as AI-driven decision-making, natural language processing, machine learning fundamentals, and AI application in various industries. Aimed at professionals seeking to advance their careers in AI, this certification equips individuals with the practical skills necessary for implementing AI strategies, optimizing workflows, and driving innovation. Achieving the AI+ Agent certification demonstrates proficiency in AI technologies, boosting credibility and enhancing job prospects in the rapidly evolving tech landscape.

Prerequisites :

- **Basic Understanding of AI Concepts –** Familiarity with core AI principles.
- **Programming Knowledge –** Proficiency in Python or similar languages.
- **Data Analysis Skills –** Ability to interpret and manipulate datasets.
- **Problem-Solving Mindset –** Analytical thinking to address AI challenges.
- **Familiarity with Machine Learning –** Understanding basic ML algorithms and techniques.

Module 1

Introduction to AI Agents

1.1 Understanding AI Agents

What are AI Agents?: AI agents are intelligent systems designed to perform tasks autonomously by processing multimodal data, reasoning, learning, and adapting. They differ from chatbots by having goal-oriented autonomy and advanced decision-making capabilities.

1.2 Anatomy and Ecosystem of AI Agents

- **Core Components of an AI Agent:** AI agents consist of four critical components: perception, reasoning, action, and learning. These components allow agents to interact with the environment, make informed decisions, and improve over time through experience.
- **Agent Ecosystems: Brief Tour of Current Platforms:** Leading frameworks such as LangChain, Auto-GPT, CrewAI, and MetaGPT offer tools to build and manage agents. These platforms enable agents to reason, plan, and collaborate in multi-agent ecosystems.
- **Factors to Consider When Choosing an AI Agent Framework:** When selecting an AI agent framework, consider factors like complexity, data privacy, ease of use, integration capabilities, scalability, and the specific needs of your project to ensure an efficient solution.

1.3 Applications, Misconceptions, and Mini Case Studies

- **Use Cases for AI Agents:** AI agents are widely applied in customer support, sales automation, software development, and more. They help streamline tasks, improve decision-making, and reduce human involvement in repetitive or complex processes.
- **Myth-Busting: What AI Agents Can and Cannot Do:** Common myths about AI agents—such as being mere chatbots or unpredictable—are debunked. AI agents are sophisticated systems capable of goal-driven actions, multi-tasking, and working within defined constraints to deliver real value.
- **Mini Case Studies – Success and Failures of AI Agents:** Success stories like AlphaFold and DoNotPay highlight the immense potential of AI agents. Conversely, failures like Microsoft's Tay emphasize the need for responsible design and careful implementation of AI technologies.

1.4 Case Study: Transforming Customer Support at Acme Retail with AI Agents

Acme Retail successfully implemented AI agents to automate routine customer support tasks, resulting in faster response times, reduced costs, and improved customer satisfaction, demonstrating the tangible benefits of AI in business.

1.5 Hands-On Exercise 1: Build a Q&A ChatBot Using Gemini + Prompt + LLM Chain in Flowise Cloud

In this hands-on exercise, learners build a dynamic Q&A chatbot using Google Gemini, Flowise Cloud's visual platform, and LLM Chain, reinforcing practical skills in integrating AI models and creating conversational agents.

Module 2

Core Concepts & Types of AI Agents

2.1 Anatomy of an AI Agent

- **Introduction:** AI agents are designed to autonomously perform tasks. The anatomy of an AI agent encompasses components like perception, reasoning, memory, and action that allow the agent to respond and interact intelligently.
- **What is Agent Architecture?**
  AI agent architecture defines how agents perceive inputs, process data, make decisions, and take actions. It integrates components like memory, reasoning, and tool use to enable intelligent, goal-driven behavior.
- **Core Capabilities:** Core capabilities of AI agents include tool use, memory retention, dialogue management, reasoning, and decision-making. These abilities empower agents to interact with users, solve problems, and adapt to various contexts autonomously.

2.2 Classification of AI Agents

- **Based on Intelligence Level:** AI agents can be classified by intelligence level, ranging from simple reflex agents to advanced learning agents. This classification reflects how agents process input, plan, and learn from experiences.

- **Based on Functional Capability:** This classification focuses on what AI agents can do, such as performing tasks autonomously, accessing APIs, using multiple tools, or automating complex workflows. Agents are designed to fulfill specific functional roles.

- **Classification Based on Application Domain:** AI agents can be classified based on the industry or domain they serve, such as customer support, healthcare, legal, or education. Domain-specific agents are optimized to handle tasks and interact within specific contexts.

2.3 Matching Agents to Use Cases

- **When to Use Which Type of AI Agent:** Choosing the right AI agent involves matching business needs with appropriate agent types. Considerations include task complexity, level of autonomy, data integration, and scalability to maximize efficiency and business impact.

- **How to Choose the Right** AI Agent: Selecting the ideal AI agent requires evaluating the business task, technological requirements, scalability, cost, and compliance considerations. This ensures a tailored, effective solution for automating workflows or solving specific challenges.

Module 3

Tools for Non-Coders

3.1 No-code and Visual Agent Platforms

- **Introduction:** No-code and visual agent platforms empower users to create intelligent agents without writing code. These platforms offer drag-and-drop interfaces to automate tasks, build chatbots, and deploy AI solutions quickly.

- **Key Features of No-Code and Visual Agent Platforms:** These platforms offer visual workflow builders, logic controls, integrations, natural language interfaces, and testing tools. Features like reusable components and easy-to-use interfaces make building intelligent agents accessible to non-technical users.

- **Key Benefits of No-Code and Visual Agent Platforms:** No-code platforms democratize technology, enabling anyone to build AI agents. Benefits include speed, flexibility, cost savings, and scalability. They also empower teams to create, test, and deploy agents quickly without coding.

3.2 Tools Overview and Setup

- **Flowise AI:** Flowise is a low-code/no-code platform that allows users to build and deploy AI workflows visually. It supports integrations with tools like LangChain and Google's LLMs for dynamic, scalable AI applications.

- **Langflow:** Langflow is an open-source visual tool built on LangChain for creating LLM-powered applications. It enables users to design, test, and deploy AI-driven workflows with minimal code, making LLM integration seamless.

- **Relevance AI:** Relevance AI is a no-code platform for automating business processes with AI agents. It allows users to create multi-agent teams and orchestrate workflows, empowering teams to automate tasks like lead enrichment and data processing.

- **Zappier AI:** Zapier is a no-code automation platform that connects over 6,000 apps. It enables users to automate workflows between apps, saving time by triggering actions like email responses or data updates with no coding.

- **Ottogrid:** Ottogrid is a no-code AI-powered spreadsheet interface that enables bulk research automation. Each cell acts as an individual AI agent, performing tasks like web scraping and document analysis to enhance data workflows.

3.3 Start building: "Your First Flow" with n8n

- **What is n8n?:** n8n is an open-source automation tool that allows users to connect apps, services, and custom logic into powerful workflows. It provides flexibility, deeper customization, and self-hosting options for complex tasks.

- **Setup: Getting Started with n8n:** n8n offers both cloud and self-hosted versions. Users can set up and create workflows to automate multi-step tasks, connect APIs, and use AI models, all with minimal code

- **Understanding & Building a Workflow in n8n:** n8n workflows consist of nodes that perform specific actions or logic. Workflows can be designed visually by connecting these nodes, making it easy to automate tasks like email responses or data analysis

3.4 Case Study: Empowering HR with AI – Building an Onboarding Assistant Without Coding

This case study showcases how an HR manager used a no-code platform to automate the employee onboarding process. The AI-powered chatbot provided personalized support, reduced manual work, and improved HR efficiency.

3.5 Hands-on Exercise

Title: Automated Employee Onboarding Agent Using n8n and Google Gemini: This hands-on exercise guides users through creating an automated onboarding workflow using Google Forms, n8n, and Google Gemini. The workflow collects data, generates personalized emails, and automates the onboarding process

Module 4

Building Simple Agents

4.1 Agent 1: AI-Powered HR Policy Assistant

- **Problem Statement:** HR policies are often stored in lengthy, unorganized formats like PDFs, making it difficult for employees to find answers quickly. This leads to inefficiency, inconsistent responses, and frustration for both staff and employees.

- **Example Scenarios:** Employees need quick answers about HR policies, like leave days or reimbursement procedures. Currently, they must sift through multiple documents or wait for HR, which wastes time and reduces productivity across the organization.

- **Goal:** To create an AI assistant using Relevance AI that can instantly retrieve answers to HR-related queries by accessing company policy documents. This will enhance employee experience, reduce HR workload, and increase productivity.

4.5 Troubleshooting and Validation of AI Agents

- **Why Troubleshooting AI Agents Matters:** AI agents rely on various interconnected components. Troubleshooting helps quickly identify and resolve issues, restoring functionality, preventing system failures, and ensuring consistent performance in production environments to maintain user trust.

- **Potential Breakdown Areas in an AI Agent:** Issues in any of the layers—such as the prompt, retrieval, reasoning, or memory layers—can lead to errors. Identifying these breakdowns early ensures swift resolution and improves agent accuracy, reliability, and user satisfaction.

- **Step-by-Step Troubleshooting Approach:** Begin by clearly defining the symptom of the issue, then inspect logs to track where the error occurred. After identifying the breakdown area, apply the appropriate fix and test the agent to ensure the issue is resolved.

- **Validation of AI Agents:** Validation ensures agents provide accurate, safe, and consistent results. This process includes cross-checking outputs, performing automated and manual tests, and stress testing to ensure the agent meets performance, safety, and reliability standards.

4.6 Share Your AI Agent

- **Direct Link Sharing: S**haring your AI agent via a direct link is a fast and easy way to make it accessible to others for testing or demos. However, it offers limited control over who can access the agent.

- **Website Embed:** Embedding your AI agent directly into a website provides a seamless user experience, making it easy to showcase products or share educational content. It requires basic HTML knowledge and access to a hosting platform.

- **Messaging Platform Integration:** Integrating your AI agent with messaging platforms like Slack or Telegram allows real-time interaction, ideal for team collaboration or community engagement. However, it requires configuring platform-specific settings and maintenance.

- **Developer Platforms & APIs:** Sharing your agent on developer platforms like GitHub or Hugging Face allows others to integrate it into their own systems. This requires providing clear documentation for setup and version control for ongoing maintenance.

4.7 Hands-on Exercise 1: Design and Implementation of an AI-Powered Research Assistant using Flowise

- **Problem Statement:** Current search engines and AI models often lack real-time data retrieval and context retention. A more integrated solution is needed to provide users with fast, accurate, and contextually aware answers to complex queries.

- **Goal:** Design a research assistant using Flowise that combines real-time data retrieval, calculations, and memory management to provide fast, context-aware answers. The assistant will use tools like Serp API and Google Gemini for enhanced performance.

- **List of Steps:** Log into Flowise, create a new chatflow, integrate tools like Serp API, Calculator, and Buffer Memory, and test the assistant to ensure it provides accurate, context-driven responses using all integrated tools.

Module 5

Multi-Tool Agents and Workflow Automation

5.1 Multi-Tool Agent

- **What is a Multi-Tool Agent?:** A Multi-Tool Agent coordinates multiple external tools, selecting and combining them based on the user's needs. It adapts to different tasks by integrating specialized tools, offering a cohesive solution to complex problems.
- **Architecture of a Multi-Tool Agent:** The architecture involves multiple components: input handling, reasoning engine, tool selection, orchestration, and memory management. These elements work together to handle dynamic, multi-step tasks by leveraging the best tools for each stage.
- **Benefits of Multi-Tool Agents:** Multi-Tool Agents offer completeness, adaptability, and scalability. They can dynamically swap tools, adjust tasks, and personalize outputs, ensuring optimized, comprehensive solutions that evolve over time as new tools are added.

5.2 Agent Chaining and Workflow Basics

- **What is Agent Chaining?:** Agent chaining involves linking multiple AI agents or tools to execute a series of tasks in a logical sequence. This structured approach enables multi-step reasoning, improving efficiency and clarity in problem-solving.
- **Types of Agent Chaining:** Agent chaining includes sequential, branching, iterative, and parallel patterns. These types manage task dependencies, dynamic conditions, repeated refinements, and simultaneous actions, offering flexibility for various workflows with different complexity levels.
- **Orchestration in Multi-Tool Agents:** Orchestration ensures that tasks within a multi-tool agent workflow are executed correctly and efficiently. It manages dependencies, error handling, and optimization, ensuring smooth transitions and results when coordinating multiple tools or agents.
- **Tools & Platforms for Agent Chaining and Orchestration:** Platforms like LangChain Hub, Make.com, AutoGPT, and LangGraph provide tools and frameworks for building and managing agent workflows, simplifying integration and offering pre-built templates for complex, multi-step automation tasks.

5.3 Managing Agent State: State, Context, and User Journey

- **Core Concept of State, Context, and User Journey:** State management in agents involves tracking goals, constraints, plans, and past actions to ensure coherent decisions. Context provides a personalized experience, while the user journey ensures transparency and trust in the agent's actions.

5.4 Prompt Engineering for Agents

- **What is Prompt Engineering?:** Prompt engineering is the art of designing clear, structured, and context-aware instructions to guide AI agents. It ensures relevant, focused, and adaptive responses by aligning the agent's output with the user's needs.
- **Why Prompt Engineering Matters for Agents?:** Effective prompt engineering enables agents to solve complex tasks, integrate external tools, and adapt to dynamic user needs. It enhances accuracy, consistency, and autonomy, making AI systems more reliable and capable of multi-step tasks.

- **Advanced Prompting Techniques for AI Agents:** Advanced techniques like Zero-Shot, Few-Shot, and Chain-of-Thought prompting refine an AI agent's reasoning, accuracy, and adaptability. These methods guide the agent through structured problem-solving, improving the quality of its outputs.

- **Prompt Parameters in AI Agents:** Prompt parameters such as temperature, top-p, max tokens, and stop sequences control the AI's creativity, response length, and focus. These settings fine-tune the agent's behavior, enabling tailored responses for diverse applications.

5.5 Multi-Agent System

- **What is a Multi-Agent System?:** A Multi-Agent System (MAS) is a network of AI agents that collaborate to solve complex tasks. By dividing tasks among specialized agents, MAS can handle large-scale, dynamic problems with greater adaptability and efficiency.

- **Single-Agent versus Multi-Agent Systems:** Single-agent systems are limited in scope, handling only specific tasks, while multi-agent systems enable collaboration, adaptability, and scalability, allowing for distributed decision-making and increased fault tolerance in complex environments.

- **Multi AI Agents Architecture:** MAS can operate in centralized or decentralized architectures. Centralized systems maintain a single knowledge base, while decentralized systems promote agent autonomy and resilience, ensuring adaptability and fault tolerance without relying on a single point of failure.

- **Real-World Applications:** MAS are used across industries like healthcare, finance, and customer service, optimizing complex tasks like treatment planning, fraud detection, and supply chain management. They also power autonomous vehicles and disaster response efforts, showcasing their versatility.

5.6 Case Study: Chaining Tools for Smarter Marketing Campaigns

- **Background:** A mid-sized e-commerce company faced fragmented tools and slow campaigns. By chaining AI, analytics, and automation tools, they improved campaign efficiency, reduced launch times, and increased customer engagement, showcasing the power of integrated workflows.

- **Key Results:** Tool chaining reduced campaign launch times by 65%, increased email open rates by 40%, and tripled conversion rates. The integrated system improved attribution accuracy and reduced operational costs, demonstrating the value of connected marketing tools.

5.7 Hands-on Exercise 1: Automating Order Tracking and Real-Time Notifications using Make.com

- **Problem Statement:** Manual order tracking and customer communication are error-prone and inefficient. By automating the process through Google Forms, Google Sheets, and Make.com, businesses can improve accuracy, reduce delays, and enhance the customer experience.

- **Goals:** The exercise aims to centralize data collection, ensure accurate order tracking, send real-time confirmation emails, and automate administrative tasks. This will save time, improve efficiency, and enhance customer satisfaction through timely communication.

- **List of Steps to be Followed:** The steps include creating an account on Make.com, building a scenario for order tracking, connecting Google Forms to capture customer orders, mapping data to Google Sheets, and sending automated email confirmations.

Module 6

Integration, Application Mapping & Deployment

6.1 Deploying Agents

- **What is Agent Deployment?:** Agent deployment involves moving an AI system from development and testing environments into production, ensuring it operates across web, mobile, and messaging platforms. It integrates workflows, handles scalability, and ensures security.
- **Key Elements of Agent Deployment:** Key elements of agent deployment include selecting the right channels for user interaction, choosing hosting environments, integrating business data, ensuring security, and setting up monitoring mechanisms to maintain performance and adaptability.

6.2 Channel Selection

- **What is Channel Selection?:** Channel selection is the process of determining where users will interact with the AI agent. The choice of platform (web, mobile, messaging, or voice) significantly impacts user engagement, adoption, and overall experience.
- **Deployment Options:** Deployment options include web applications, mobile apps, messaging platforms, and voice interfaces. Each channel has its strengths and suits specific interaction patterns, with web apps offering easy access and voice interfaces providing hands-free use.
- **Best Practices:** Start with the most popular channel for your audience, tailor the user experience to that channel, maintain consistency across platforms, and plan for future multi-channel expansion as your user base grows.

6.3 Hosting Environment

- **What is a Hosting Environment?:** A hosting environment defines where the AI agent operates after deployment, with options such as on-premise, cloud, or SaaS. This choice affects scalability, cost, security, and compliance requirements for the deployment.
- **Best Practices for Hosting Environment:** Match the hosting model with business needs—on-premise for control, cloud for scalability, or SaaS for quick deployment. Consider compliance, cost, security, and the ability to scale as demand increases over time.

6.4 Data Integration

- **Spreadsheets & CSV Files:** Spreadsheets and CSV files are simple to integrate for small-scale deployments. They are ideal for quick prototypes, storing structured data like project lists or customer records, but have limitations in scalability and real-time access.
- **SQL Databases:** SQL databases support structured data and complex queries, offering robust scalability and real-time access. They're ideal for enterprise-grade applications where high volumes of data and reliable performance are essential for business operations.
- **Vector Databases:** Vector databases like Chroma, Pinecone, and Weaviate enable AI agents to perform semantic search and retrieval-augmented generation (RAG), allowing them to deliver context-aware results by understanding the meaning behind data, not just keywords.
- **NoSQL Databases:** NoSQL databases handle semi-structured data, providing flexibility, scalability, and high performance. They are ideal for AI agents working with large, dynamic datasets like product catalogs, user preferences, or event streams, supporting real-time access.

6.5 Security Setup

- **Authentication & Authorization:** Authentication verifies user identity, while authorization controls access to resources. Best practices include multi-factor authentication, role-based access, and least-privilege principles to ensure agents only perform actions allowed for authenticated users.
- **API Key & Credential Management:** Effective API key management involves securely storing keys, rotating them regularly, and applying the least-privilege principle to minimize the risk of exposure or misuse. Monitoring and auditing API usage is essential for security.
- **Public vs Private Deployments:** Public deployments are open to anyone but require safeguards like rate limiting, while private deployments limit access to authenticated users. Deciding between them depends on the balance between accessibility and data protection.
- **Data Privacy & Compliance:** Compliance with regulations like GDPR, HIPAA, and CCPA is crucial. AI agents should respect data privacy, provide users with control over their data, and ensure data protection measures are in place to avoid legal risks.

6.6 Monitoring & Updates

- Monitoring ensures agent performance by tracking accuracy, uptime, and user feedback, while regular updates and error handling maintain reliability. Continuous improvement through model retraining, security audits, and feedback analysis enhances adaptability, accuracy, and compliance.

6.7 Application Mapping

- **Which Agent for Which Scenario?:** Mapping agents to use cases ensures their effectiveness. For example, sales agents qualify leads, HR agents automate queries, and support agents resolve tickets. Correct application mapping aligns AI functionality with business goals.
- **Mini Case Study:** Case studies illustrate how agents are applied in real-world scenarios, like Moveworks automating HR and IT queries, Salesloft enhancing sales engagement, and Ada improving customer support. These examples show the transformative power of AI agents.

Module 7

Monitoring, Guardrails & Responsible AI

7.1 Observability Basics

- **What is Observability in AI Agents?:** Observability in AI agents involves tracking metrics, events, logs, and traces (MELT data) to explain agent behavior. This allows teams to assess performance, debug issues, and ensure ethical compliance by analyzing system actions.
- **Key Components of Observability in AI Agents:** Key components of observability include logging, metrics, tracing, and events. These elements capture data about AI interactions, decision-making, and system performance, enabling transparency, debugging, and accountability for continuous improvement.

7.2 Performance Evaluation: Key Metrics

- **What Are Agent Evaluation Metrics?:** Agent evaluation metrics measure how well AI agents perform tasks, interact with users, and make decisions. These metrics assess functionality, ethical compliance, efficiency, and user experience, ensuring agents meet design intent and organizational goals.

- **Why Agent Evaluation Metrics Are Important:** Evaluation metrics are crucial for ensuring AI agents perform efficiently, ethically, and as intended. They highlight areas for improvement, confirm ethical behavior, track task completion, and ensure agents remain aligned with business objectives.

7.3 Guardrails: Preventing Misuse & Ensuring Safe Outputs

- **What Are AI Guardrails?:** AI guardrails are safety mechanisms designed to prevent AI agents from producing harmful, biased, or irrelevant outputs. They ensure agents remain focused, ethical, and aligned with intended purposes by limiting their actions and responses.

- **Why Do Guardrails Matter?:** Guardrails protect AI systems from hallucinations, biased responses, privacy breaches, and non-compliant actions. They ensure safe, reliable, and ethical behavior, preventing risks and safeguarding against misuse while maintaining user trust and legal compliance.

- **How Do Guardrails Work?:** Guardrails operate by evaluating both inputs and outputs at various stages, ensuring safety. They include filters, correctors, safety checks, and human-in-the-loop mechanisms, all working together to prevent unsafe, biased, or harmful AI behaviors.

- **Types of AI Guardrails:** Guardrails include appropriateness, hallucination, compliance, alignment, validation, privacy, and action guardrails. Each type protects against specific risks, ensuring agents operate safely, ethically, and within legal constraints, aligning with their intended roles.

- **Input & Output Guardrails for Agents:** Input guardrails protect against malicious instructions, privacy violations, and off-topic requests. Output guardrails ensure responses are accurate, safe, and aligned with ethical standards, preventing harmful, biased, or illegal content from being delivered.

- **What is Responsible Agent Behaviour?:** Responsible agent behavior ensures AI agents act ethically, safely, and within their intended scope. This includes respecting privacy, making ethical decisions, ensuring compliance, maintaining transparency, and escalating sensitive cases to human oversight when needed.

7.4 Responsible AI

- **What is Responsible AI?:** Responsible AI involves designing and deploying AI systems that are safe, fair, transparent, and accountable. It ensures that AI respects human values, complies with regulations, and contributes positively to societal well-being.

- **Core Principles of Responsible AI:** The core principles of Responsible AI include fairness, transparency, accountability, privacy, safety, and human-centric design. These principles guide AI development, ensuring systems are ethical, compliant, and aligned with user and societal needs.

- **Operationalizing Responsible AI:** Operationalizing Responsible AI involves embedding ethics into every stage of the AI lifecycle, from design to deployment. It includes using ethical frameworks, continuous monitoring, and tools like model cards and datasheets to ensure accountability and compliance.

7.5 Mini-Case: Failure and Recovery in Agent Deployments

- **Case Study 1: Replit's Coding Agent – Catastrophic Deletion & Recovery:** In this case study, Replit's AI coding agent accidentally deleted an entire live database due to lack of guardrails. The company's swift recovery involved backup restoration and stronger safeguards to prevent future issues.

- **Case Study 2: Anthropic's "Project Vend" – Vending Agent Gone Rogue:** Anthropic's vending agent, "Claudius," mispriced items and misinterpreted supply-demand relationships, highlighting AI's unpredictability. The recovery process involved learning from the incident, improving guardrails, and conducting further research to enhance agent behavior.

7.6 Real-world Failures

- **Case Study 1: Replit's Coding Agent – Catastrophic Deletion & Recovery:** Replit's coding agent caused a catastrophic database deletion but recovered swiftly by restoring backups and introducing stronger system controls. This case emphasizes the importance of guardrails, observability, and recovery mechanisms in AI deployments.
- **Case Study 2: Anthropic's "Project Vend" – Vending Agent Gone Rogue:** In "Project Vend," Anthropic's AI agent failed to perform expected tasks, mispricing items and misunderstanding basic economics. The recovery strategy focused on analyzing the incident, improving oversight, and learning from the rogue behavior.

7.7 Peer Sharing: How to Present and Discuss Agent Logs/Results

- **Why Peer Sharing Matters:** Peer sharing sessions foster early detection of issues, transparency, and collective intelligence. They enhance accountability and promote continuous learning, ensuring AI agents improve through shared insights and collaborative problem-solving.

- **Who Should Be Involved in Peer Sharing:** Effective peer sharing involves a cross-functional team, including product owners, engineers, data analysts, compliance officers, and support staff. This diverse team ensures a comprehensive review of agent behavior from technical, user, and compliance perspectives.

- **Preparing for a Peer Sharing Session:** To ensure productive peer sharing, teams should prepare summary reports, metrics snapshots, curated traces, and guardrail logs. These structured materials provide actionable insights for improving agent performance and enhancing safety protocols.

- **Running the Session:** In peer sharing sessions, teams should follow a structured approach: framing objectives, reviewing metrics, discussing logs, analyzing failures, and defining action items. This ensures focused, productive discussions aimed at improving agent behavior and performance.

Module 7

Capstone Project – Design Your Own Intelligent Agent

8.1 Capstone Project 1: Smart Personal AI Assistant

- **Problem Statement:** The problem is organizing daily tasks and managing schedules efficiently. A personal AI assistant can automate these tasks using natural language processing (NLP) to interpret user input, set reminders, and handle scheduling.

- **Objectives:** Design a personal AI assistant capable of processing natural language commands, managing tasks like scheduling, reminders, and Q&A, integrating APIs for real-time updates, and saving user data for future interactions.

8.2 Capstone Project 2: Smart Lead Engagement - From Email to Personalized Outreach

- **Problem Statement:** Sales teams struggle with parsing inbound lead emails that are unstructured and lack context. Automating lead parsing, enrichment, and personalized outreach through AI can significantly reduce manual effort and improve conversion rates.

- **Objectives:** Build a sales support agent using Zapier to automate email parsing, lead enrichment, scoring, and personalized email generation. Store lead data in Google Sheets and trigger responses to improve sales efficiency and engagement.

8.3 Capstone Project 3: Education Tutor Agent

- **Problem Statement:** Students often lack personalized guidance outside the classroom. An AI tutor that adapts explanations to individual learning styles can provide real-time support, ensuring better understanding and engagement in subjects like science or mathematics.

- **Objectives:** Design an AI tutor capable of providing personalized step-by-step explanations, quizzes, and adaptive learning paths. Integrate real-time student queries, store responses in a knowledge base, and offer tailored suggestions to improve learning outcomes.

8.4 HR Knowledge Bot

- **Use Case & Scenario:** An HR knowledge bot answers employee questions on policies, benefits, and leaves. By leveraging Relevance AI for RAG-based responses and Slack integration, the bot enhances HR efficiency and provides consistent answers.

- **Tools & Technologies:** Relevance AI is used for policy-based responses. Slack integration ensures employees can access the bot within company channels. The bot leverages a knowledge base of HR policies and workflows to handle queries and escalate unresolved issues.

8.5 Customer Service Agent

- **Use Case & Scenario:** This AI-powered customer service agent handles Tier-1 queries like shipping and refunds, retrieving responses from a product knowledge base. It escalates unresolved issues to human agents, improving efficiency and customer satisfaction.

- **Tools & Technologies:** Relevance AI powers the FAQ knowledge base. Integration with platforms like Google Sheets or CRM systems ensures that the agent can log interactions, track tickets, and manage queries. The workflow also uses escalation via Zapier.

8.6 Healthcare Triage Bot

- **Use Case & Scenario:** This AI bot helps patients by suggesting preliminary steps based on symptoms. It uses predefined symptom guidelines to assess risk levels and escalate high-risk cases to medical professionals, improving early-stage care.

- **Tools & Technologies:** Langflow builds the chat flow, while Relevance AI pulls from vetted medical information. Google Sheets or Airtable stores patient data, and notifications alert healthcare providers when further intervention is required.