



Objectifs	Concevoir, développer, intégrer, sécuriser, déployer et superviser des solutions d'intelligence artificielle sur Microsoft Azure. Mettre en œuvre des architectures back-end évolutives combinant services de données, bases vectorielles, conteneurs, messagerie événementielle, fonctions serverless, sécurité, observabilité et pratiques de cycle de vie applicatif.
Participants	Développeurs cloud et back-end, ingénieurs IA, consultants techniques, ingénieurs DevOps et architectes souhaitant développer des solutions IA natives du cloud Azure et préparer la certification Microsoft Certified: Azure AI Cloud Developer Associate.
Prérequis	Maîtrise pratique de Python. Connaissance des principes Azure, des API REST, de Git et des SDK. Notions de Docker, de bases de données, de messagerie asynchrone et de sécurité cloud. Une expérience de Visual Studio Code et de la ligne de commande Azure est recommandée.
Moyens pédagogiques	1 poste par participant - 1 Vidéo projecteur - Support de cours fourni à chaque participant - Ateliers Individuels - Modalités d'évaluation : Ateliers (TP) pendant tout le long de la formation et Evaluation des acquis tout au long de la formation.
Méthodes pédagogiques	Exposés interactifs et démonstrations - Ateliers individuels et en groupe - Échanges d'expériences et de bonnes pratiques
Type de formation	Formation présentielle ou distancielle, selon les besoins et les contraintes des participants
Tarif inter-entreprise	3900 € HT
Durée	5 jours - 35 heures
Modalités et délais d'accès	Formations accessibles sous 15 à 20 jours suite à votre demande. Un entretien initial doit être prévu avant votre entrée en formation.

Code : NCI_9VYAYHL

Mis à jour le : 28 octobre 2026

Programme :

Module 1 : Concevoir et préparer une solution IA cloud sur Azure

Analyser les exigences fonctionnelles, techniques, de sécurité et d'exploitation d'une solution IA.

Sélectionner les services Azure, les SDK et les modèles d'architecture adaptés.

Préparer l'environnement de développement avec Visual Studio Code, Python, Git, Azure CLI et les SDK Azure.

Gérer la configuration applicative, les identités managées et les secrets avec Azure Key Vault.

Structurer le code, les dépendances, les tests et la gestion des erreurs d'une application cloud.

Appliquer les principes de résilience, de scalabilité, de coût et d'IA responsable.

Lab / Atelier : Initialiser la solution fil rouge : créer les ressources, configurer l'identité et les secrets, structurer le dépôt Python, connecter les SDK Azure et valider un premier appel sécurisé.

Module 2 : Développer la couche de données et la recherche vectorielle

Concevoir une couche de données adaptée aux applications IA et aux charges de travail distribuées.

Développer avec Azure Cosmos DB for NoSQL : modèles de données, requêtes, indexation et gestion du débit.

Mettre en œuvre Azure Database for PostgreSQL et l'extension pgvector pour la similarité vectorielle.

Utiliser Azure Managed Redis pour la mise en cache, les performances et les scénarios vectoriels.

Générer, stocker et interroger des embeddings pour une recherche sémantique.

Appliquer les patterns de génération augmentée par récupération, les filtres de métadonnées et la gestion de la pertinence.

Lab / Atelier : Construire un service de recherche sémantique : ingérer un jeu de documents, générer les embeddings, indexer les vecteurs, comparer les stratégies de recherche et exposer une API de récupération.

Module 3 : Conteneuriser, déployer et mettre à l'échelle les composants IA

Créer des images Docker sécurisées et reproductibles pour des services Python.

Publier et versionner les images dans Azure Container Registry.

Déployer une API IA dans Azure Container Apps et gérer les révisions.

Comprendre les composants d'Azure Kubernetes Service utiles à l'hébergement de services IA.

Configurer les paramètres, secrets, identités, probes et règles de mise à l'échelle.

Choisir entre Container Apps, AKS et autres options d'hébergement selon le scénario.

Lab / Atelier : Conteneuriser et déployer l'API IA : créer l'image, la publier dans ACR, déployer sur Azure Container Apps, configurer



l'identité managée, les secrets et la mise à l'échelle, puis exécuter des tests de charge.

Module 4 : Intégrer les services back-end et les traitements événementiel

Concevoir des traitements asynchrones et découplés pour des pipelines IA.

Implémenter des files, sujets et abonnements avec Azure Service Bus.

Publier et traiter des événements avec Azure Event Grid.

Développer des fonctions serverless avec Azure Functions, déclencheurs et liaisons.

Gérer les reprises, les doublons, les files de lettres mortes et l'idempotence.

Intégrer les services de données, la couche IA et les composants back-end dans un workflow cohérent.

Lab / Atelier : Créer un pipeline IA événementiel : soumettre un document, publier un événement, déclencher un traitement serverless, appeler le service IA, persister le résultat et traiter les erreurs via une file dédiée.

Module 5 : Sécuriser, superviser et industrialiser la solution

Sécuriser les accès avec Microsoft Entra ID, RBAC, identités managées et principe du moindre privilège.

Protéger les secrets, les connexions réseau et les données sensibles.

Instrumenter l'application avec Azure Monitor, Application Insights et OpenTelemetry.

Exploiter les journaux, métriques, traces distribuées et requêtes KQL pour diagnostiquer les incidents.

Mettre en place des contrôles de santé, alertes, tableaux de bord et stratégies de reprise.

Préparer le déploiement automatisé, la validation opérationnelle et l'amélioration continue.

Lab / Atelier : Finaliser la solution fil rouge - appliquer les contrôles de sécurité, activer la télémétrie distribuée, diagnostiquer un incident simulé, définir les alertes et présenter l'architecture ainsi que les choix d'exploitation.