

Nouveau Sessions garanties disponibles

# Formation IA Agentique : concevoir et développer des agents LLM autonomes

Réf. SF34531

Maîtrisez les design patterns agentiques pour construire des systèmes IA capables de raisonner, utiliser des outils et coordonner plusieurs agents spécialisés !

|              |                                   |                                 |                        |
|--------------|-----------------------------------|---------------------------------|------------------------|
| 2 j<br>Durée | 2 000 € HT<br>Inter / participant | 4 200 € HT<br>Intra (≤12 pers.) | 31<br>Sessions à venir |
|--------------|-----------------------------------|---------------------------------|------------------------|

## Objectifs pédagogiques

- ✓ Distinguer un agent IA autonome d'un chatbot classique et identifier les cas d'usage adaptés
- ✓ Concevoir des workflows de planification et d'exécution de tâches complexes
- ✓ Implémenter le pattern de réflexion pour améliorer itérativement les sorties d'un agent
- ✓ Développer un système multi-agents avec des architectures adaptées (réseau, superviseur, hiérarchique)
- ✓ Connecter un agent à des outils externes (APIs, bases de données, exécution de code)
- ✓ Évaluer les performances d'un système agentique et optimiser ses coûts et sa latence

## Public visé & prérequis

**Public visé** Data scientists, Développeurs, Ingénieurs Big Data

**Prérequis** - Maîtrise de Python (niveau intermédiaire) - Connaissance des APIs REST (requêtes, réponses, authentification) - Bases en IA générative (avoir utilisé un LLM via API ou interface)

## Programme détaillé

### Introduction à l'IA agentique et aux workflows autonomes

- Agents vs chatbots : comprendre ce qui distingue un agent autonome d'un assistant conversationnel classique
- Degrés d'autonomie : du prompt unique au système multi-agents entièrement orchestré
- Design patterns agentiques : les 4 motifs fondamentaux (réflexion, tool use, planification, multi-agent)
- Cas d'usage sectoriels : service client, analyse de données, automatisation de workflows, génération de contenu
- Décomposition de tâches : identifier les étapes d'un workflow et évaluer ce qui peut être délégué à un agent

#### Activités pratiques :

- Exploration d'un agent de recherche autonome pour comprendre le fonctionnement d'un workflow agentique
- Analyse comparative d'un prompt direct vs un workflow agentique sur une même tâche

### Réflexion et amélioration itérative des sorties

- Pattern de réflexion : l'agent évalue ses propres sorties et itère pour améliorer la qualité
- Boucle réflexive : architecture d'un cycle génération → critique → régénération
- Auto-critique vs feedback externe : deux approches complémentaires pour corriger les erreurs
- Évaluation de l'impact : mesurer concrètement le gain de qualité apporté par la réflexion

#### Activités pratiques :

- Implémentation d'un workflow de génération de graphiques avec boucle de réflexion
- Amélioration de la génération de requêtes SQL par itération réflexive

## Utilisation d'outils et intégration d'APIs

- Function calling : permettre à un agent d'invoquer des fonctions Python comme des outils
- Syntaxe et déclaration d'outils : définir les schémas, paramètres et descriptions pour le LLM
- Exécution de code : donner à l'agent la capacité d'écrire et d'exécuter du code dynamiquement
- Intégration d'APIs externes : connecter un agent à des bases de données, services web et moteurs de recherche
- MCP (Model Context Protocol) : protocole standardisé pour exposer des outils aux LLMs de manière interopérable

### Activités pratiques :

- Transformation de fonctions Python en outils utilisables par un agent
- Construction d'un assistant email capable de lire, trier et répondre via des outils

## Planification et exécution de tâches complexes

- Décomposition de problèmes : faire raisonner l'agent pour découper une tâche complexe en étapes exécutables
- Pattern Plan & Execute : séparer la phase de planification de la phase d'exécution
- Re-planification dynamique : adapter le plan en cours d'exécution selon les résultats intermédiaires
- Pattern ReAct (Reasoning + Acting) : combiner raisonnement et actions dans une boucle unifiée
- Parallélisation et dépendances : orchestrer l'exécution de tâches indépendantes et gérer les séquences

### Activités pratiques :

- Développement d'un agent de service client capable de planifier et exécuter des actions (consultation de base, remboursement, escalade)
- Implémentation d'un agent planificateur avec exécution de code

## Systèmes multi-agents et orchestration

- Agent unique vs multi-agents : quand et pourquoi distribuer le travail entre plusieurs agents spécialisés
- Architectures multi-agents : réseau (peer-to-peer), superviseur centralisé, hiérarchie d'agents
- Patterns de communication : échange de messages, mémoire partagée, protocoles de coordination
- Mémoire et état : gestion d'état par agent, mémoire partagée vs isolée, persistance
- Graphes d'agents : modéliser les interactions entre agents comme un graphe d'exécution

### Activités pratiques :

- Construction d'une équipe de recherche multi-agents (collecte, analyse, synthèse) avec coordination par superviseur
- Implémentation d'un système multi-agents coordonné avec gestion de la mémoire partagée

## Évaluation, optimisation et bonnes pratiques

- Évaluations (evals) : concevoir des métriques pour mesurer la performance d'un système agentique
- Évaluations composant par composant : isoler et tester chaque maillon du workflow
- Analyse d'erreurs : diagnostiquer les échecs et prioriser les améliorations
- Optimisation des coûts et de la latence : réduire les appels LLM inutiles, paralléliser, gérer les timeouts
- Human-in-the-loop : intégrer des points de validation humaine dans les workflows critiques

### Activités pratiques :

- Mise en place d'évaluations automatisées sur un workflow agentique existant
- Analyse d'erreurs et optimisation coût/latence d'un système multi-agents

## Prochaines sessions

| DATE       | LIEU       | STATUT           |
|------------|------------|------------------|
| 01/12/2026 | à distance | Session garantie |
| 26/01/2027 | Lyon       | Sur confirmation |
| 26/01/2027 | Grenoble   | Sur confirmation |
| 26/01/2027 | Paris      | Sur confirmation |
| 26/01/2027 | Lille      | Sur confirmation |

| DATE       | LIEU       | STATUT           |
|------------|------------|------------------|
| 26/01/2027 | à distance | Sur confirmation |

## Pourquoi choisir Sparks Formation ?

20 ans d'expertise IT, des formateurs experts et des sessions garanties contre l'annulation.

20 ans  
d'expertise

1 068+  
formations

11 178  
apprenants formés/an

+67  
Net Promoter Score

### Avis des participants

4,9/5 — 190 avis

**Anne-Lise C.** — Très bonne adaptation du formateur au public et au niveau d'attente.

**Marie G.** — La formatrice était très pédagogue et bienveillante ! Elle a su nous partager sa passion et donner envie de pratiquer à la suite Voilà bien longtemps que je...

### Votre formateur

**Camilo R.** — Formateur

Camilo intervient notamment sur l'IA appliquée aux métiers et sa gouvernance, ainsi que sur l'ingénierie des LLM et l'IA générative. Son activité de formation a débuté en 2024. La conception...

### Ils nous font confiance

Cette année notre filière Intelligence artificielle & Data Science a accompagné plus de 544 entreprises dont :

SOPRA STERIA THALES GLOBAL SERVICES ARTELIA AXA ASSISTANCE FRANCE GRANT THORNTON FRANCE AKKODIS DIGITAL SAS  
MEDIABRANDS CEGEDIM COVAGE INFRA SNCF - E.SNCF SOLUTIONS +534 autres

### Financement & certifications

Cette formation est éligible aux financements OPCO et FNE-Formation. Nos équipes vous accompagnent dans le montage de votre dossier.



### Réservez votre place dès maintenant

20 ans d'expertise · 1 000+ formations · NPS +67 · réponse sous 24h ouvrées

**0 805 950 800**

demande@sparks-formation.com