# *Maitriser le développement avec des agents IA*

## *Développement agentique : Spec-driven development, MCP et skills au service d'un SDLC sous contrôle*

### DESCRIPTION

La généralisation du fonctionnement agentique dans les assistants de code modifie en profondeur le métier du développeur. Les assistants de code ne se contentent plus de proposer, ils prennent en charge la réalisation d'objectifs de manière autonome ou supervisée: planification, enchaînement des actions, modification des bases de code, exécution de tests, débogage… Le développeur n'écrit quasiment plus de code : il pilote le développement, prend les décisions stratégiques. L'offre est vaste, les promesses réelles mais l'usage demande de la rigueur et un apprentissage.. Comment choisir les bons outils ? Pour quels cas d'usage ? Et surtout, comment les utiliser en gardant la maîtrise du flux de développement ainsi que la capacité à faire évoluer le code produit ? l'IA code, mais le développeur crafte.

L'apparition des agents, leur capacité à agir via des MCP et l'amélioration de la qualité des modèles a fait émerger de nouvelles pratiques pour développer avec l'IA : formalisation d'exigences par et pour l'agent, création de “skills”, artefacts intermédiaires, apparitions de nouvelles approches (BMAD, spec driven development).

Cette formation explore, à travers de nombreuses mises en pratique, le nouveau paysage du développement agentique : ses outils, ses cas d'usage et les méthodes pour en exploiter tout le potentiel. Elle vise à permettre l'adoption d'une posture de développeur augmenté — éclairé, productif, exigeant, responsable et pleinement maître de ses choix technologiques.

### OBJECTIFS PEDAGOGIQUES

- Expliquer les concepts fondamentaux des LLMs, décrire l'écosystème des outils d'IA générative et distinguer les différents modes d'interaction possibles
- Expliquer les principes du mode agentique et distinguer les différences entre prompt simple et approche par compétences (skills)
- Expérimenter l'approche spec-driven en développant un cas d'usage concret basé sur des spécifications
- Expliquer le rôle des MCP (Model Context Protocols) dans une architecture d'IA
- Adopter une posture active face aux productions IA : contrôle, séquençage du travail, relecture, amélioration. Expérimentation sur des cas plus complexes

**Stage pratique**
Qualité du logiciel - Software Craftsmanship

Code :
**AI4DV**

Durée :
**2 jour(s) (14,00 heures)**

Exposés : **30 %**
Cas pratiques : **40 %**
Echanges d’expérience : **30 %**

**Inter-entreprises :**
Prochaines sessions disponibles sur notre site web.
Tarif : 1 920,00 € HT / participant

**Intra-entreprise :**
Tarifs et dates sur demande.

- Adopter une approche critique et collaborative, orientée craft, afin d'évaluer la qualité des usages et d'améliorer continuellement ses pratiques.
- Intégrer dans sa pratique les limites juridiques et sécuritaires à l'usage de ces outils

**PUBLIC CIBLE**

- Développeur
- Tech Lead
- Architecte

**PRE-REQUIS**

- Maîtrise d'un langage de programmation structuré comme Java, Javascript ou Python

**METHODE PEDAGOGIQUE**
Formation avec apports théoriques, échanges sur les contextes des participants et retours d'expérience pratique des formateurs, complétés de travaux pratiques et de mises en situation.

**PROFIL DES INTERVENANTS**

Cette formation est dispensée par un·e ou plusieurs consultant·es d'OCTO Technology ou de son réseau de partenaires, expert·es reconnus des sujets traités.

Le processus de sélection de nos formateurs et formatrices est exigeant et repose sur une évaluation rigoureuse leurs capacités techniques, de leur expérience professionnelle et de leurs compétences pédagogiques.

**MODALITÉS D'ÉVALUATION ET FORMALISATION À L'ISSUE DE LA FORMATION**
L'évaluation des acquis se fait tout au long de la session au travers des ateliers et des mises en pratique. Afin de valider les compétences acquises lors de la formation, un formulaire d'auto-positionnement est envoyé en amont et en aval de celle-ci. Une évaluation à chaud est également effectuée en fin de session pour mesurer la satisfaction des stagiaires et un certificat de réalisation leur est adressé individuellement.

**PROGRAMME PEDAGOGIQUE DETAILLE**

**Jour 1**

***OUVERTURE DE SESSION***

- Accueil des stagiaires et tour de table des attentes
- Présentation du déroulé de la formation

***INTRODUCTION AU DÉVELOPPEMENT ASSISTÉ PAR IA***

- Rappels sur les LLMs et les assistants de code
- Présentation des diverses méthodes d'interaction (completion, chat, agent) au travers de l'outil qui sera utilisé pendant la session
- Familiarisation avec le set-up de l'outillage
- Développement du cas d'utilisation en mode agentique avec l'outil mis à disposition
- Premiers enseignements et Partage d'expérience collectifs sur les résultats obtenus

***ÉCOSYSTÈME ET OUTILS***

- Panorama des assistants de code agentiques (Github, Copilot, Claude Code, OpenCode, etc.)
- Coûts, modèles économiques, limites techniques

***DÉVELOPPEMENT AGENTIQUE***

- Comprendre les skills et les prompts
- Présentation de l'approche spec-driven development (planification, découpage en tâche, validation, testing)
- Rédaction et organisation d'informations contextuelles réutilisables liées au projet (ex : contexte métier, utilisateurs/personas, contraintes techniques, spécifications techniques …)
- Rédaction et organisation de directives réutilisables liées (skills, workflows, agents) à des opérations (ex : création de code front, code back, création de test, création de diagramme, analyse de bug…

***OUTILLAGE AGENTIQUE***

- Gestion de directives et d'informations contextuelles
- Skills, workflows et agents dans les différents outils
- Assistance au workflow de développement (versionning, revue de code, …)

**Jour 2**

***LE MODÈLE MCP***

- Genèse du modèle
- Fonctionnement du modèle
- Mise en pratique : "Interactions au travers d'un MCP"

***RETOUR D'EXPÉRIENCE***

- Retour d'expérience sur des cas rencontrés en mission sur

l'implantation de l'approche spec-driven, de l'utilisation de MCP et de la mise en oeuvre de skills et agents

***MARKDOWN DRIVEN FRAMEWORK***

- Fonctionnement, intérêts et produits
- Mise en pratique : "développer avec un framework de type spec-kit"

***SÉCURITÉ, CADRE LÉGAL ET ÉTHIQUE***

- Enjeux de sécurité liés au code produit
- Souveraineté et confidentialité
- Propriété intellectuelle
- Enjeux de sécurité liés aux outils utilisés (MCP en particulier)
- Enjeux éthiques : biais, impact écologique

***IA & COÛTS***

- Type de tarification
- Notion de cache
- Stratégies d'optimisation

***IA ET MODERNISATION***

- Quel type d'application, quel type de rénovation
- Reverse engineering et approches spécifiques à la rénovation
- Montée de version de framework ou langage

***CLÔTURE DE SESSION***

- Revue des concepts clés et bilan
- Échange autour des questions et réponses additionnelles

**Accessibilité**

L'inclusion est sujet important pour OCTO Academy.
Nos référent·es sont à votre disposition pour faciliter l'adaptation de votre formation à vos besoins spécifiques.
Pour les contacter : academy.accessibilite@octo.com