

AWS – Développement piloté par les spécifications avec Kiro

Durée: 2 jours

Public cible: Professionnel TI

À qui s'adresse cette formation ?

Cette formation s'adresse aux :

- développeurs et ingénieurs logiciels;
- responsables techniques qui adoptent des méthodes de développement assistées par l'IA;
- équipes de développement qui adoptent des pratiques de codage agentique;
- ingénieurs DevOps qui intègrent des agents d'IA aux pipelines de livraison.

Objectifs de la formation

À la fin de cette formation, vous serez en mesure de :

- mettre en œuvre des flux de développement piloté par les spécifications afin de planifier, de concevoir et de réaliser des fonctionnalités et des correctifs;
- configurer et contrôler le comportement des agents au moyen de fichiers de pilotage (steering files), de fournisseurs de contexte, des modes d'exécution Autopilot et Supervised, ainsi que de sous-agents personnalisés (custom subagents);
- étendre les capacités des agents à l'aide de serveurs Model Context Protocol (MCP) et de Kiro Powers afin d'intégrer des outils, des services et une expertise spécialisée externes;
- automatiser et déployer à grande échelle les flux de développement à l'aide des agent hooks, des agent skills, de fichiers de pilotage partagés et de sous-agents personnalisés;
- créer et partager des capacités agentiques réutilisables en développant des Kiro Powers, des compétences d'agents (agent skills) et des sous-agents personnalisés adaptés aux flux de travail d'une équipe;
- appliquer des mécanismes de gouvernance d'entreprise pour la liste d'autorisation du registre MCP, la gouvernance des modèles et la gestion des extensions.

Description sommaire

Développement piloté par les spécifications avec Kiro est une formation de deux jours qui s'adresse aux développeurs, aux ingénieurs logiciels et aux responsables techniques qui adoptent des pratiques de développement agentique assisté par l'IA.

Cette formation présente les principes du codage agentique et explique comment les mettre en pratique avec Kiro, l'environnement de développement intégré (IDE) agentique conçu pour le développement logiciel.

Les participants apprendront à planifier et à mettre en œuvre des fonctionnalités à l'aide des Feature Specs et des Bugfix Specs de Kiro, à contrôler le comportement des agents au moyen de fichiers de pilotage (steering files) et de modes d'exécution, ainsi qu'à étendre leurs capacités à l'aide de serveurs Model Context Protocol (MCP) et de Kiro Powers.

La formation couvre également l'automatisation des flux de travail au moyen des agent hooks, l'intégration d'une expertise réutilisable grâce aux agent skills et aux custom subagents, ainsi que la configuration des projets afin de favoriser la collaboration en équipe et la gouvernance d'entreprise. Des laboratoires pratiques sont intégrés aux différents modules tout au long des deux journées. Le jour 1, les participants développent progressivement une application météo en utilisant les différentes fonctionnalités de Kiro à mesure qu'elles sont présentées. Le jour 2, ils déploient et enrichissent une application de microblogage préconfigurée, puis la configurent afin d'en permettre l'utilisation en équipe.

Contenu du plan de cours

Jour 1

Module 1 : Introduction au développement agentique avec Kiro

- Le paradigme du codage agentique et les principales fonctionnalités de Kiro
- Les cinq principes d'un développement agentique efficace
- Kiro IDE comparativement à Kiro CLI
- Les sessions Vibe, les sessions Spec et la sélection des modèles

Module 2 : Développement piloté par les spécifications avec les Feature Specs

- Le flux de travail en trois phases : exigences, conception et tâches
- Le flux de travail Requirements-First et la notation EARS
- Le flux de travail Design-First : conception de haut niveau et de bas niveau
- Validation de l'exactitude au moyen de tests basés sur les propriétés (property-based testing)

Module 3 : Pilotage, contexte et contrôle des agents

- Les fichiers de pilotage (steering files) et le contexte fondamental
- Les fichiers de pilotage personnalisés et leurs modes d'inclusion : always, fileMatch, manual et auto
- La portée de l'espace de travail (workspace) et la portée globale
- Les fournisseurs de contexte et la gestion des sessions
- Les modes d'exécution Autopilot et Supervised
- Les commandes et les outils approuvés (trusted commands et trusted tools)

Module 4 : Bugfix Specs

- Quand utiliser les Bugfix Specs
- La phase d'analyse d'un correctif : comportement actuel, comportement attendu et comportement qui doit demeurer inchangé
- L'analyse de la cause fondamentale et la conception de la solution
- Les tests basés sur les propriétés pour prévenir les régressions

Laboratoire du jour 1

Fondements du développement piloté par les spécifications avec Kiro

Jour 2

Module 5 : Serveurs MCP et intégration d'outils

-

Les principes fondamentaux du Model Context Protocol (MCP) et son architecture client-serveur

- La configuration des serveurs MCP à l'échelle de l'espace de travail et de l'utilisateur
- La gestion des serveurs, des outils et de l'approbation automatique (auto-approve)
- L'utilisation des outils MCP dans les conversations et le dépannage

Module 6 : Kiro Powers

- Le problème de la surcharge contextuelle et la façon dont les Kiro Powers permettent d'y remédier
- La structure des répertoires d'un Power : POWER.md, mcp.json et steering
- L'installation et l'utilisation des Powers à partir de l'IDE et de kiro.dev
- La création et le partage de Powers personnalisés

Module 7 : Agent Hooks, Skills et Custom Subagents

- Agent hooks : déclencheurs, actions et création
- Agent skills : ensembles d'instructions portables reposant sur la divulgation progressive (progressive disclosure)
- Custom subagents : autorisations d'outils à portée définie et délégation
- Comparaison entre Skills, Steering et Powers

Module 8 : Collaboration en équipe et gouvernance d'entreprise

- Configuration partagée de l'espace de travail au moyen du répertoire .kiro
- Organisation des fichiers de pilotage et conventions d'équipe pour la révision des spécifications
- Gouvernance d'entreprise : liste d'autorisation du registre MCP, gouvernance des modèles, registre des extensions et outils Web

Laboratoire du jour 2

Approche et méthode pédagogique

La formation repose sur une approche structurée combinant exposés théoriques, démonstrations et travaux pratiques en environnement AWS. Les notions sont mises en application au fur et à mesure de leur apprentissage dans AWS Kiro, permettant aux participants d'expérimenter concrètement les pratiques de développement agentique et de développement piloté par les spécifications.

Au cours des laboratoires, les participants développent, déploient et font évoluer des applications tout en utilisant les principales fonctionnalités de Kiro. Cette approche favorise le transfert des apprentissages vers des situations réelles de développement logiciel.

Prérequis

Il est recommandé que les participants possèdent :

- une bonne maîtrise d'au moins un langage de programmation;
- une connaissance pratique du contrôle de versions avec Git;
- une compréhension de base des API RESTful et des concepts liés aux applications infonuagiques;
- des connaissances fondamentales des concepts d'IA générative, notamment les grands modèles de langage (LLM), les jetons (tokens) et l'inférence