

## Developing on Hyperledger Fabric Blockchain

**Durée: 2 Jours**    **Réf de cours: U67927G**

### Résumé:

Cette formation Hyperledger dirigée par un instructeur est conçue pour les développeurs et ceux qui veulent faire une plongée complète dans Hyperledger Fabric v1.4.

Cette formation a été créée pour vous guider à travers le développement, le test et le déploiement du Chaincode pour un réseau Hyperledger Fabric, en particulier pour le Chaincode écrit en Golang (le langage Chaincode original de Fabric). En outre, en tant que développeur d'applications, vous apprendrez à écrire et à préparer des applications clientes en utilisant le kit de développement standard le plus mature de Hyperledger Fabric, NodeJS. Il y a des cas d'utilisation, des preuves de concepts, ainsi que des travaux pratiques interactifs sur les concepts.

### Objectifs pédagogiques:

- Comprendre pourquoi et où la blockchain est nécessaire
- Explorer les principaux composants de Blockchain
- Apprendre à connaître Hyperledger Fabric et la structure de l'architecture Hyperledger.
- Découvrez les caractéristiques du modèle Fabric, notamment le chaincode, les SDK, le Ledger, la sécurité et les services d'adhésion.
- Effectuer des tests complets sur l'écriture de chaincode
- Explorer l'architecture de l'Hyperledger Fabric
- Comprendre et réaliser des exercices approfondis sur le démarrage du réseau.
- Effectuez des exercices complets pour intégrer/développer une application avec Hyperledger Fabric en exécutant un contrat intelligent.
- Créer des applications sur Hyperledger Fabric

### Pré-requis:

Ce cours est très technique par nature et nécessite que l'étudiant soit à l'aise avec le codage. Pour se préparer au cours, tous les étudiants DOIVENT : Se familiariser avec la structure et les objectifs des composants Hyperledger Fabric.

- 
-

## Contenu:

- Blockchain Basics (Overview)
- Hyperledger Fabric Development Environment
- Knowing the Difference: Composer
- Chaincode Use Cases
- Chaincode Basics
- Golang Shim Development
- Databases for the Developer
- Chaincode Dev. Deployment and Interactions
- Clients ; SDK Development: Fabric-Network
- Clients ; SDK Development: Fabric-Client Interactions
- Logging and Monitoring

Lab: Setting up the Developer Environment

- Blockchain Basics (Overview)
- Hyperledger Fabric Development Environment
- Knowing the Difference: Composer
- Chaincode Use Cases
- Chaincode Basics
- Golang Shim Development
- Databases for the Developer
- Chaincode Dev. Deployment and Interactions
- Clients ; SDK Development: Fabric-Network
- Clients ; SDK Development: Fabric-Client Interactions
- Logging and Monitoring

Lab: Writing the Chaincode Structure

- Blockchain Basics (Overview)
- Hyperledger Fabric Development Environment
- Knowing the Difference: Composer
- Chaincode Use Cases
- Chaincode Basics
- Golang Shim Development
- Databases for the Developer
- Chaincode Dev. Deployment and Interactions
- Clients ; SDK Development: Fabric-Network
- Clients ; SDK Development: Fabric-Client Interactions
- Logging and Monitoring

Lab: Creating Rich Queries

- Blockchain Basics (Overview)
- Hyperledger Fabric Development Environment
- Knowing the Difference: Composer
- Chaincode Use Cases
- Chaincode Basics
- Golang Shim Development
- Databases for the Developer
- Chaincode Dev. Deployment and Interactions
- Clients ; SDK Development: Fabric-Network
- Clients ; SDK Development: Fabric-Client Interactions
- Logging and Monitoring

Lab: Packaging ; Deploying the Chaincode

- Blockchain Basics (Overview)
- Hyperledger Fabric Development Environment
- Knowing the Difference: Composer
- Chaincode Use Cases
- Chaincode Basics
- Golang Shim Development
- Databases for the Developer
- Chaincode Dev. Deployment and Interactions
- Clients ; SDK Development: Fabric-Network
- Clients ; SDK Development: Fabric-Client Interactions
- Logging and Monitoring

Lab: SDK Development Pt. 1 Writing User Persistence Info

Lab: SDK Development Pt. 2 Transacting and Querying

- Blockchain Basics (Overview)
- Hyperledger Fabric Development Environment
- Knowing the Difference: Composer
- Chaincode Use Cases
- Chaincode Basics
- Golang Shim Development
- Databases for the Developer
- Chaincode Dev. Deployment and Interactions
- Clients ; SDK Development: Fabric-Network
- Clients ; SDK Development: Fabric-Client Interactions
- Logging and Monitoring

## Autres moyens pédagogiques et de suivi:

- Compétence du formateur : Les experts qui animent la formation sont des spécialistes des matières abordées et ont au minimum cinq ans d'expérience d'animation. Nos équipes ont validé à la fois leurs connaissances techniques (certifications le cas échéant) ainsi que leur compétence pédagogique.
- Suivi d'exécution : Une feuille d'émargement par demi-journée de présence est signée par tous les participants et le formateur.
- Chaque participant, à l'issue de la formation, répond à un questionnaire de satisfaction qui est ensuite étudié par nos équipes pédagogiques en vue de maintenir et d'améliorer la qualité de nos prestations.

Délais d'inscription :

- Vous pouvez vous inscrire sur l'une de nos sessions planifiées en inter-entreprises jusqu'à 5 jours ouvrés avant le début de la formation sous réserve de disponibilité de places et de labs le cas échéant.
- Votre place sera confirmée à la réception d'un devis ou ""booking form"" signé. Vous recevrez ensuite la convocation et les modalités d'accès en présentiel ou distanciel.
- Attention, si vous utilisez votre Compte Personnel de Formation pour financer votre inscription, vous devrez respecter un délai minimum et non négociable fixé à 11 jours ouvrés.