

Blockchain Developer Ethereum Training

Durée: 3 Jours Réf de cours: U67945G

Résumé:

Apprenez les composants d'Ethereum et comment configurer un réseau public Ethereum. Ce cours vous présentera Solidity et vous montrera comment construire une application de base et développer des contrats intelligents grâce à des exercices pratiques complets. Vous passerez également en revue web3 et le framework Truffle avant de relier le tout en construisant un projet web complet avec Ethereum. Cette formation a été élaborée avec soin et est dispensée par des experts qui vous guideront à travers des sujets approfondis, notamment les étapes d'intégration/développement d'une application avec Ethereum. Il y a des cas d'usage, des exemples, ainsi que des laboratoires pratiques interactifs.

Public visé:

Ce cours est conçu pour les programmeurs et les développeurs expérimentés qui souhaitent approfondir l'écriture de Smart Contracts et la création d'applications qui interagissent avec eux. Apprenez les composants d'Ethereum et comment configurer un réseau public Ce cours vous présentera Solidity et vous montrera comment construire une application de base et développer des contrats intelligents grâce à des exercices pratiques complets. Vous passerez également en revue web3 et le framework Truffle avant de relier le tout en construisant un projet web complet avec Ethereum.

Objectifs pédagogiques:

- Ce que vous apprendrez dans ce cours :
 - Les bases de la blockchain et les types de blockchain.
 - Les principaux composants de la Blockchain Ethereum
 - Le langage de programmation d'Ethereum, Solidity
- Les contrats intelligents
- Développement d'une DApp v1.1
- The Ethereum Development Training Course is designed for those seeking an in-depth understanding and development experience of the Ethereum Blockchain platform. Students will participate in approximately 50% programming lab time providing practical experience, enhancing their knowledge and existing skill set. Due to the technical programming lab content covered in this course, it is not recommended for those without programming knowledge and experience.

Pré-requis:

Ce cours est très technique et nécessite que les stagiaires soient à l'aise avec la programmation.

- Nécessaire: Connaissances de base en JavaScript / HTML
- Recommandé : Connaissances de base enC ++ / Java, data types, git repositories

Contenu:

Introduction

- Background
- Introductions
- Overview
- Scheduling
- Lab Environment

What is Blockchain?

- The History of Decentralized Ledgers
- Blockchain Mechanics
- Assets in Blockchain
- The Blockchain Ledger
- Type of Blockchain Transactions
- Blockchain as Workflow

The History of Blockchain

- Bringing DLT into the Modern Age
- What is a Block?
- How are Blocks Chained?
- Bitgold and Quest for Digital Currency
- Bitcoin
- Ethereum
- Hyperledger

How Does Blockchain Work?

- Benefits and Drawbacks of Blockchain
- Cryptography: 2 way functions
- Cryptography: Hashes
- Databases vs. Blockchain

Blockchain Use Cases and Real-World Implementations

- Blockchain Use Case Symptoms
- Blockchain Use Case Analysis and Scorecarding
- Good Blockchain Use Case Patterns
- Real World Implementations
- The Web 3.0 Revolution

Group Consensus Models

- Proof of Work Deep Dive
- The is The Nonce?
- Hacking the Ledger
- Costs of a 51% Attack
- Group Consensus Models and Game Theory

Proof of Stake and the Future of Ethereum

- How Does PoS Work?
- PoS vs PoW
- Ethereum 2.0
- Sharding
- How does PoW fit in?
- Group Consensus vs Participant Consensus

Ethereum Application Architecture

DevOps Considerations

- Ethereum Smart Contract Lifecycle
- The Self-Destruct Function
- Agile in the World of Blockchain?
- The Project Team
- Blockchain Projects vs Traditional Projects

Security Considerations

- Protecting Self-Destruct
- Code Visibility
- DevOps Vulnerabilities
- Common Blockchain Exploits

Ethereum 2.0

- Ethereum 2.0 - What's Different
- Ethereum Shards

Lab Environment Setup and Configuration

- Downloading and Installing VirtualBox
- Setting Up Your Environment
- Configuring Your Environment

Lab Overview

Lab 1: Introduction

Lab 2: The Remix IDE

Lab 3: MetaMask and the Test Networks

Lab 4: Infura.io

Lab 5: Web3.js

Lab 6: Events

Lab 7: Function Modifiers

Lab 8: Mappings and Structs

Lab 9: Inheritance

Lab 10: Deploying to Live Networks

Lab 11: Bonus Lab - The Pet Shop Tutorial

- Decentralized Application Layers
- Technologies and Components of Each Layer
- Development Tools, Frameworks, and Utilities

Sample Application Walkthrough

- Sample Application Use Case Walkthrough
- End User Solution Review
- Solution Code Review: User Interface
- Solution Code Review: Integration Layer
- Solution Code Review: Smart Contract

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.
- Modalités d'évaluation : le participant est invité à s'auto-évaluer par rapport aux objectifs énoncés.
- 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.