Blockchain, développer sur Tezos

Cours Pratique de 4 jours - 28h
Réf : BKT - Prix 2025 : 2 920 HT

La blockchain Tezos se distingue des précédentes générations (Bitcoin, Ethereum) par trois caractéristiques principales : la gouvernance permettant l'auto-évaluation du protocole, le consensus Liquid Proof Of Stake et le développement des smart contracts avec le langage Michelson autorisant la vérification formelle.

OBJECTIFS PÉDAGOGIQUES

À l'issue de la formation l'apprenant sera en mesure de :

Appréhender les concepts de la blockchain Tezos

Modéliser et créer un smart contract en Ligo

Mettre en place les tests unitaires sur un smart contract

Déployer un smart contract avec Michelson

Interagir avec un smart contract déployé

S'approprier des modèles simples de smart contracts

TRAVAUX PRATIQUES

Pédagogie déductive basée sur des échanges, des exemples, des démonstrations.

LE PROGRAMME

dernière mise à jour : 01/2024

1) La blockchain Tezos

- Introduction à la blockchain et au consensus.
- Architecture et enjeux.
- Gouvernance en chaîne.
- Discussion vérification formelle.
- Les cas d'usage par segment (finance, énergie, logistique, agroalimentaire...).
- Les interfaçages possibles de la blockchain avec d'autres technologies comme le big data, l'Internet des objets, l'IA.

2) Interactions avec la blockchain Tezos

- Installation.
- Lancement d'un nœud.
- Outil de ligne de commande.
- Interactions par HTTP.

Travaux pratiques : Création d'un nœud en sandbox et transactions.

3) Les langages Michelson et Ligolang

- Michelson : smart contract (paramètres, storage, code).
- Michelson : langage et commandes, point d'accès, typage.

Travaux pratiques : Exécution d'un exemple simple avec Michelson.

4) Le langage Ligolang (Pascaligo)

- Les bases de la programmation fonctionnelle.
- Le langage.
- Storage, entrypoint, transaction.
- Outil de transpilation.

PARTICIPANTS

Développeurs, architectes, ingénieurs concepteurs.

PRÉREQUIS

Connaissances de l'environnement Linux (ssh, bash) et d'au moins un langage de programmation.

COMPÉTENCES DU FORMATEUR

Les experts qui animent la formation sont des spécialistes des matières abordées. Ils ont été validés par nos équipes pédagogiques tant sur le plan des connaissances métiers que sur celui de la pédagogie, et ce pour chaque cours qu'ils enseignent. Ils ont au minimum cinq à dix années d'expérience dans leur domaine et occupent ou ont occupé des postes à responsabilité en entreprise.

MODALITÉS D'ÉVALUATION

Le formateur évalue la progression pédagogique du participant tout au long de la formation au moyen de QCM, mises en situation, travaux pratiques…
Le participant complète également un test de positionnement en amont et en aval pour valider les compétences acquises.

MOYENS PÉDAGOGIQUES ET TECHNIQUES

• Les moyens pédagogiques et les méthodes d'enseignement utilisés sont principalement : aides audiovisuelles, documentation et support de cours, exercices pratiques d'application et corrigés des exercices pour les stages pratiques, études de cas ou présentation de cas réels pour les séminaires de formation.
• À l'issue de chaque stage ou séminaire, ORSYS fournit aux participants un questionnaire d'évaluation du cours qui est ensuite analysé par nos équipes pédagogiques.
• Une feuille d'émargement par demi-journée de présence est fournie en fin de formation ainsi qu'une attestation de fin de formation si le stagiaire a bien assisté à la totalité de la session.

MODALITÉS ET DÉLAIS D'ACCÈS

L'inscription doit être finalisée 24 heures avant le début de la formation.

ACCESSIBILITÉ AUX PERSONNES HANDICAPÉES

Pour toute question ou besoin relatif à l'accessibilité, vous pouvez joindre notre équipe PSH par e-mail à l'adresse psh-accueil@orsys.fr.

- Fonction anonyme.
- Bonnes pratiques.

Travaux pratiques : Simulation d'un smart contract Ligo avec manipulation de counter smart contract, smart contrat de vote.

5) Tests unitaires

- Paramètres d'appel et storage.
- Gestion des exceptions.
- Mise en place et utilisation de PyTezos.

Travaux pratiques : Manipulation avec counter smart contract.

6) Déploiement de smart contracts

- Préparation des paramètres et storage.
- Comptes et faucet.
- Simulation du déploiement.
- Déploiement/baking.

Travaux pratiques : Premier déploiement.

7) Invocation de smart contract

- Préparation des paramètres d'invocation.
- Exécution.

Travaux pratiques : Invocation et accès au storage.

8) Vérification formelle et modèle de smart contract avec Tezos

- Coq et Mi-Cho-Coq.
- Démarche et modélisation pour la vérification formelle.
- La librairie SmartPy.
- Interactions entre smart contracts et "polymorphism".

Travaux pratiques : Exemple de vote. Multisig et proxy.

LES DATES

CLASSE À DISTANCE
2025 : 16 sept., 25 nov.

PARIS
2025 : 09 sept., 18 nov.