

SCHOOL FOR MATE

LIVRET PÉDAGOGIQUE

Blockchain appliquée au développement : cas d'usage concrets

CT03 · Module de spécialisation

Ce module de 21 heures forme les développeurs web à intégrer des fonctionnalités blockchain dans une application réelle. Le parcours suit le cycle complet d'un smart contract : conception éco-responsable, développement sécurisé, tests, déploiement et interfaçage avec une application web conforme aux standards OWASP, RGPD et RGAA.

VOLUME HORAIRE	COURS	BLOCS
21 h	6	3

01 · PRÉSENTATION

La formation en bref

Intitulé	Blockchain appliquée au développement : cas d'usage concrets
Code	CT03
Type	Module de spécialisation
Certificateur	Hors répertoire
Durée	21 h, 6 cours
Modalité	Formation à distance (e-learning), tutorat assuré par l'organisme de formation
Formats	SCORM 1.2 / 2004, xAPI, sources éditables
Mise à jour	Novembre 2026

Le métier et les compétences visées

Le développeur d'applications intégrant la blockchain conçoit, programme et livre des smart contracts qui automatisent l'exécution d'accords sans intermédiaire central. Il traduit un besoin fonctionnel en modélisations et spécifications techniques, en tenant compte de la sobriété des traitements on-chain. Il développe des contrats conformes aux standards de sécurité (SCSVS), les teste de façon automatisée, puis les déploie sur une blockchain cible. Il relie enfin ces contrats à une application web existante grâce à des bibliothèques spécialisées, en garantissant la sécurité applicative, la protection des données personnelles et l'accessibilité numérique. Cette compétence élargit le métier de développeur à des infrastructures décentralisées qui redéfinissent les protocoles d'échange et de confiance.

Public visé

- Développeurs web front-end, back-end ou full-stack souhaitant intégrer des fonctionnalités blockchain
- Développeurs d'applications en poste justifiant d'au moins un an d'expérience en programmation web
- Architectes logiciels chargés d'évaluer ou de concevoir des architectures intégrant une blockchain
- Développeurs en reconversion ou en montée en compétences vers les technologies décentralisées
- Lead developers devant encadrer un projet applicatif reposant sur des smart contracts

Prérequis

- Justifier d'un an d'expérience professionnelle en programmation web (prérequis du référentiel RS6604)
- Maîtriser un langage de programmation web, idéalement JavaScript ou TypeScript
- Savoir utiliser un gestionnaire de versions (Git) et un terminal en ligne de commande
- Disposer d'un poste permettant l'installation d'un environnement de développement (Node.js, éditeur de code, navigateur avec extension de portefeuille)

Objectifs pédagogiques

À l'issue du parcours, l'apprenant est capable de :

- Analyser un besoin métier pour déterminer la pertinence et le périmètre d'une intégration blockchain
- Formaliser les modélisations et spécifications techniques d'un smart contract en appliquant les bonnes pratiques d'éco-conception
- Développer un smart contract en Solidity conforme aux spécifications validées
- Appliquer les exigences du standard SCSVS pour prévenir les vulnérabilités courantes des smart contracts
- Implémenter des tests unitaires et d'intégration du smart contract à l'aide de bibliothèques de tests
- Définir et exécuter une procédure de déploiement manuelle ou automatisée sur une blockchain cible
- Interfacer une application web existante avec un smart contract à l'aide de bibliothèques spécialisées
- Intégrer les exigences OWASP, RGPD et RGAA dans l'interface entre l'application et la blockchain
- Réaliser des tests fonctionnels de l'interaction entre l'application et la blockchain, puis traiter les dysfonctionnements relevés

02 · FIL ROUGE PÉDAGOGIQUE

Traçalis

IG

Inès Garnier

Développeuse full-stack, puis référente technique blockchain

Éditeur de logiciels SaaS de traçabilité pour les filières agroalimentaires

Traçalis est une PME lyonnaise de 35 salariés qui édite une plateforme web de traçabilité utilisée par des coopératives agricoles. Son principal client, la coopérative viticole fictive Les Coteaux de Valfleury, souhaite délivrer à chaque lot de bouteilles un certificat d'origine infalsifiable, consultable par les distributeurs et les consommateurs via un QR code. La direction technique confie ce projet à Inès Garnier, développeuse full-stack depuis trois ans, experte en TypeScript mais novice en blockchain. Elle doit concevoir un smart contract de certification de lots, le sécuriser, le tester, le déployer puis l'intégrer à l'application existante, tout en respectant la sobriété énergétique, la protection des données des producteurs et l'accessibilité de l'interface. Son responsable, Marc Delorme, CTO, valide chaque étape lors de revues techniques.

Chaque cours se termine par une mise en situation dans cette entreprise : l'apprenant accompagne Inès dans des situations professionnelles réalistes, de plus en plus complexes, jusqu'à la préparation des épreuves.

La progression

1. Bloc M1 : Inès analyse le besoin de la coopérative, choisit une blockchain cible et rédige les spécifications techniques éco-conçues du contrat de certification des lots, qu'elle présente en revue à Marc Delorme.
2. Bloc M2 : Inès code le smart contract en Solidity, applique la checklist SCSVS, puis construit une suite de tests automatisés qui révèle et corrige une faille de contrôle d'accès avant tout déploiement.
3. Bloc M3 : Inès déploie le contrat sur un réseau de test puis en production via un script automatisé, l'interface avec l'application Traçalis de façon sécurisée, conforme au RGPD et accessible, et valide le tout par des tests fonctionnels de bout en bout.

03 · ARCHITECTURE

Organisation du parcours

Le parcours est construit sur les compétences visées : chaque bloc regroupe les cours qui préparent à son évaluation.

N°	Bloc et cours	Durée
M1 · Comprendre la blockchain et concevoir un smart contract éco-responsable		**6 h**
1	Fondamentaux de la blockchain et cas d'usage applicatifs	3 h
2	Conception technique d'un smart contract et éco-conception	3 h
M2 · Développer, sécuriser et tester un smart contract		**8 h**
3	Développement d'un smart contract sécurisé en Solidity selon le standard SCSVS	4 h 30
4	Tests automatisés d'un smart contract	3 h 30
M3 · Déployer un smart contract et l'interfacer avec une application web		**7 h**
5	Déploiement d'un smart contract sur une blockchain cible	3 h
6	Interfaçage d'une application web avec la blockchain : sécurité, RGPD, accessibilité et tests fonctionnels	4 h
Total du parcours		**21 h**

Répartition des modalités pédagogiques

Chaque sous-chapitre s'ouvre sur une vidéo de 3 à 5 minutes. La vidéo représente 30 % du volume horaire, complétée par du cours texte enrichi, des infographies, des études de cas, des mises en situation et des QCM.



- Vidéos : 6 h 30
- Cours texte et infographies : 6 h 30
- Études de cas et mises en situation : 5 h
- QCM et évaluations : 3 h

04 · PROGRAMME DÉTAILLÉ

Contenu des cours

M1 · 6 h

Comprendre la blockchain et concevoir un smart contract éco-responsable

Cours 1 · Fondamentaux de la blockchain et cas d'usage applicatifs

3 h

OBJECTIFS

- Expliquer le fonctionnement d'une blockchain, de ses mécanismes de consensus et des smart contracts
- Évaluer la pertinence d'une solution blockchain face à un besoin métier donné

ÉVALUATION FORMATIVE

QCM de 15 questions sur les notions fondamentales et étude de cas courte : justification écrite du choix d'une architecture blockchain

CONTENUS

- Registre distribué, blocs, hachage et immutabilité : les principes fondamentaux
- Mécanismes de consensus (preuve de travail, preuve d'enjeu) et impacts énergétiques
- Blockchains publiques, privées et de consortium, réseaux compatibles EVM et solutions de couche 2
- Comptes, transactions, frais de gas et portefeuilles
- Rôle et limites des smart contracts : automatisation, immuabilité, oracles
- Cas d'usage concrets : traçabilité, certification, jetons, gouvernance
- Grille de décision : quand la blockchain est utile, quand elle ne l'est pas

Fil rouge · Inès reçoit le cahier des charges de la coopérative Les Coteaux de Valfleury et doit démontrer à Marc Delorme, à l'aide d'une grille de décision argumentée, pourquoi une blockchain publique compatible EVM répond au besoin de certificats infalsifiables.

Cours 2 · Conception technique d'un smart contract et éco-conception

3 h

OBJECTIFS

- Formaliser les modélisations (diagrammes de cas d'utilisation, de séquence, d'états) d'un smart contract
- Rédiger des spécifications techniques intégrant les bonnes pratiques d'éco-conception

ÉVALUATION FORMATIVE

Exercice pratique : production d'un dossier de spécifications (modélisations et choix d'éco-conception) évalué à l'aide d'une grille de critères

CONTENUS

- Du besoin fonctionnel aux user stories d'un smart contract
- Modélisation UML adaptée : acteurs, cas d'utilisation, diagrammes de séquence et machines à états
- Définition des données on-chain et off-chain, structures de stockage et événements
- Rôles, permissions et modèle de contrôle d'accès dès la conception
- Éco-conception : minimisation du stockage et des calculs on-chain, optimisation du gas, choix du réseau
- Prise en compte des standards existants (ERC-20, ERC-721, ERC-1155) et de la réutilisation de bibliothèques auditées
- Rédaction et validation d'un dossier de spécifications techniques

Fil rouge · Inès modélise le contrat de certification des lots de vin : elle décide de ne stocker on-chain qu'une empreinte cryptographique de chaque certificat et rédige les spécifications qu'elle soumet à la revue technique de Marc Delorme.

M2 · 8 h

Développer, sécuriser et tester un smart contract

Cours 3 · Développement d'un smart contract sécurisé en Solidity selon le standard SCSVS

4 h 30

OBJECTIFS

- Développer un smart contract en Solidity conforme aux spécifications validées
- Appliquer les exigences du standard SCSVS pour prévenir les vulnérabilités courantes

ÉVALUATION FORMATIVE

Exercice de code guidé avec correction commentée et QCM sur les vulnérabilités et exigences SCSVS

CONTENUS

- Mise en place de l'environnement de développement (Node.js, Hardhat ou Foundry, éditeur de code)
- Syntaxe Solidity : types, variables d'état, fonctions, modificateurs, événements et erreurs personnalisées
- Héritage et utilisation de bibliothèques auditées (OpenZeppelin Contracts)
- Présentation du Smart Contract Security Verification Standard (SCSVS) et de ses catégories d'exigences
- Vulnérabilités courantes : réentrance, contrôle d'accès défaillant, dépassements, manipulation de l'horodatage, déni de service
- Bonnes pratiques : modèle checks-effects-interactions, principe du moindre privilège, pause d'urgence
- Analyse statique et revue de code sécurité d'un contrat
- Optimisation du gas en cohérence avec les choix d'éco-conception

Fil rouge · Inès code le contrat de certification à partir de ses spécifications puis passe sa première version au crible de la checklist SCSVS : elle identifie qu'une fonction d'émission de certificat est appelable par n'importe quel compte et met en place un contrôle d'accès par rôles.

Cours 4 · Tests automatisés d'un smart contract

3 h 30

OBJECTIFS

- Implémenter des tests unitaires et d'intégration d'un smart contract à l'aide de bibliothèques de tests
- Interpréter les résultats des tests pour détecter et corriger des dysfonctionnements techniques et logiques

ÉVALUATION FORMATIVE

Travaux pratiques : rédaction d'une suite de tests atteignant un taux de couverture défini, accompagnée d'un court rapport d'anomalies corrigées

CONTENUS

- Stratégie de test d'un smart contract : unitaires, intégration, cas limites et cas d'erreur
- Écriture de tests avec les bibliothèques de l'écosystème (Hardhat avec Mocha et Chai, ou Foundry)
- Réseau local de test, comptes simulés et manipulation du temps et des blocs
- Tests des contrôles d'accès, des événements émis et des messages d'erreur
- Tests par fuzzing et tests d'invariants
- Mesure de la couverture de code et du coût en gas
- Diagnostic, correction et test de non-régression

Fil rouge · Inès écrit la suite de tests du contrat de certification ; un test d'invariant révèle qu'un même lot peut être certifié deux fois, elle corrige la logique puis ajoute un test de non-régression avant la revue avec Marc Delorme.

M3 · 7 h

Déployer un smart contract et l'interfacer avec une application web

Cours 5 · Déploiement d'un smart contract sur une blockchain cible

3 h

OBJECTIFS

- Définir une procédure de déploiement manuelle ou automatisée adaptée à la blockchain cible
- Rendre un smart contract disponible et vérifiable pour les utilisateurs et les applications tierces

ÉVALUATION FORMATIVE

Exercice pratique : rédaction d'une procédure de déploiement et réalisation d'un déploiement vérifié sur un réseau de test

CONTENUS

- Environnements de déploiement : réseau local, réseau de test, réseau principal
- Configuration des réseaux, fournisseurs de nœuds RPC et gestion sécurisée des clés privées
- Procédure de déploiement manuelle : étapes, contrôles et traçabilité
- Scripts de déploiement automatisés et intégration dans une chaîne CI/CD
- Vérification et publication du code source sur un explorateur de blocs
- Mise à disposition de l'ABI et de l'adresse du contrat aux applications tierces
- Stratégies d'évolution : contrats évolutifs (proxy), migration et documentation de livraison

Fil rouge · Inès rédige la procédure de déploiement du contrat de certification, l'exécute d'abord sur un réseau de test puis automatise le déploiement dans la chaîne d'intégration continue de Traçalis, en isolant les clés dans un coffre de secrets.

Cours 6 · Interfaçage d'une application web avec la blockchain : sécurité, RGPD, accessibilité et tests fonctionnels

4 h

OBJECTIFS

- Interfacer une application web existante avec un smart contract à l'aide de bibliothèques spécialisées
- Intégrer les exigences OWASP, RGPD et RGAA dans les fonctionnalités blockchain de l'application
- Mettre en œuvre des tests fonctionnels de l'interaction entre l'application et la blockchain

ÉVALUATION FORMATIVE

Projet de synthèse : intégration d'un contrat dans une application web fournie, évaluée par une grille couvrant le fonctionnement, la sécurité, la conformité RGPD, l'accessibilité et les tests fonctionnels

CONTENUS

- Bibliothèques d'interfaçage (ethers.js, viem) : lecture d'état, envoi de transactions, écoute d'événements
- Connexion des portefeuilles utilisateurs et gestion des états de transaction dans l'interface
- Sécurité applicative : risques OWASP appliqués aux applications décentralisées (signature, validation des entrées, dépendances)
- RGPD et blockchain : immuabilité et droit à l'effacement, minimisation, stockage off-chain et empreintes
- Accessibilité numérique selon le RGAA : messages de transaction, retours d'état, navigation au clavier
- Tests fonctionnels de bout en bout de l'interaction application et blockchain
- Traitement des dysfonctionnements relevés et documentation

Fil rouge · Inès intègre le contrat à la plateforme Traçalis : elle conçoit la page de vérification accessible via QR code, s'assure qu'aucune donnée personnelle des viticulteurs n'est inscrite on-chain et automatise un test de bout en bout qui simule l'émission puis la vérification d'un certificat.

05 · CONFORMITÉ AU RÉFÉRENTIEL

Couverture des compétences

Correspondance entre les compétences visées et les cours du parcours.

Compétence	Bloc	Cours
C1. Réaliser la conception technique d'un smart contract (modélisations et spécifications éco-conçues)	M1	1, 2
C2. Développer un smart contract conforme aux spécifications et aux standards de sécurité SCSVS	M2	3
C3. Livrer un smart contract sur une blockchain cible via une procédure de déploiement manuelle ou automatisée	M3	5
C4. Interfacer une application web avec une blockchain dans le respect de l'OWASP, du RGPD et du RGAA	M3	6
C5. Implémenter les tests du smart contract et les tests fonctionnels de l'interaction application et blockchain	M2	4, 6

Évaluations formatives

Chaque cours se conclut par une évaluation formative alignée sur la capacité attestée qu'il travaille : QCM de vérification des connaissances, exercices de code avec correction commentée, travaux pratiques de test et de déploiement, études de cas issues du fil rouge Traçalis. Des QCM intermédiaires jalonnent également les sous-chapitres afin de permettre au stagiaire de mesurer sa progression en continu. Le parcours se termine par un projet de synthèse mobilisant l'ensemble du cycle de vie d'un smart contract. Les grilles d'évaluation fournies permettent au formateur de l'organisme de formation d'objectiver les acquis et de tracer la progression de chaque stagiaire, conformément aux exigences Qualiopi.

Modalités de certification

Bloc	Modalité d'évaluation
M1, M2, M3	Les modalités d'évaluation de la certification RS6604 ne sont pas détaillées dans les données de référence transmises. Elles sont définies par le certificateur SIMPLON.CO et doivent être consultées sur la fiche officielle France Compétences RS6604 (certification enregistrée jusqu'au 26/04/2029). L'organisme de formation reste seul responsable de l'inscription des candidats et de l'organisation de la certification auprès du certificateur.

Le parcours couvre l'intégralité des cinq capacités attestées du référentiel RS6604 (C1 à C5), chacune étant travaillée dans au moins un cours et matérialisée par une production concrète : dossier de spécifications, smart contract sécurisé, suite de tests, procédure de déploiement, application interfacée. Le fil rouge Traçalis reproduit un projet complet de bout en bout, ce qui entraîne le stagiaire à produire et à justifier des livrables proches de ceux attendus en situation professionnelle. Le projet de synthèse final et les grilles d'évaluation par capacité permettent à l'organisme de formation d'identifier les points à consolider avant la présentation à la certification.

06 · MOYENS ET PERSPECTIVES

Moyens pédagogiques et débouchés

Moyens pédagogiques et techniques

- Plateforme e-learning en marque blanche aux couleurs de l'organisme de formation, accessible 24 h/24 sur ordinateur et tablette
- Une vidéo de 3 à 5 minutes par sous-chapitre, représentant 30 % du volume horaire, complétée par des cours textes enrichis
- Infographies, schémas d'architecture et fiches mémo téléchargeables (checklist SCSVS, procédure de déploiement, points de contrôle RGPD et RGAA)
- Études de cas et mises en situation construites autour du fil rouge Traçalis
- Dépôts de code de démarrage et corrigés commentés pour les exercices pratiques (Solidity, scripts de test et de déploiement, application web à interfacer)
- QCM interactifs avec correction immédiate et grilles d'évaluation par capacité attestée
- Guide d'installation de l'environnement de développement et utilisation de réseaux de test gratuits, sans engagement financier en cryptomonnaie

Débouchés et évolutions

- Développeur d'applications intégrant des fonctionnalités blockchain
- Développeur de smart contracts
- Développeur full-stack spécialisé en applications décentralisées
- Architecte logiciel intégrant des briques blockchain
- Référent technique blockchain au sein d'une équipe de développement

Accessibilité

Contenus sous-titrés en français, contrastes conformes WCAG AA, navigation au clavier sur les écrans interactifs. L'organisme de formation qui diffuse le parcours adapte les modalités d'accompagnement aux personnes en situation de handicap avec son référent handicap.

Livrables fournis à l'organisme de formation

- Package SCORM 1.2 / 2004 et xAPI, compatible avec les principaux LMS
- Sources éditables des cours, scripts vidéo et banques de QCM
- Livret apprenant et grilles d'évaluation
- Licence marque blanche à vie, stagiaires illimités, maintenance 24 mois

School For Mate
On crée vos formations, vous n'avez plus qu'à les diffuser.
contact@schoolformate.com · catalogue-2026-sfm.netlify.app

La délivrance de la certification relève exclusivement du certificateur et de l'organisme de formation habilité. Document non contractuel, susceptible d'évoluer avec le référentiel.