

5BLOC - Développement d'une DApp

Introduction.....	1
Définition du Cas d'Usage Choisi.....	2
1. Objectif.....	2
2. Fonctionnalités Clés.....	2
3. Technologies Utilisées.....	2
4. Scénario d'Utilisation.....	3
5. Conclusion.....	3
Choix de Conception et Respect des Contraintes Métiers.....	4
1. Choix Technologiques.....	4
1.1. Blockchain et Smart Contracts.....	4
1.2. Stockage d'Images et Documents.....	4
1.3. Frontend et Interaction Utilisateur.....	4
2. Respect des Contraintes Métiers.....	5
2.1. Achat/Vente de Propriétés.....	5
2.2. Règles de Cooldown et Limitations.....	5
2.3. Stockage décentralisé des fichiers.....	5
3. Sécurité et Robustesse.....	5
3.1. Protection des Transactions.....	5
3.2. Tests et Simulations.....	5
4. Conclusion.....	6

Introduction

Le projet Web3 Monopoly vise à créer une plateforme de gestion et d'échange de propriétés sur la blockchain, inspirée du jeu Monopoly. Ce système repose sur des NFTs (Non-Fungible Tokens) pour représenter chaque propriété et utilise IPFS (InterPlanetary File System) pour stocker les images et documents associés.

Ce rapport présente le cas d'usage choisi et les choix techniques faits pour assurer scalabilité, sécurité et respect des contraintes métiers.

Définition du Cas d'Usage Choisi

1. Objectif

L'objectif du projet est de permettre aux utilisateurs de :

Acheter, vendre, modifier et échanger des propriétés de manière décentralisée.

Stocker de manière sécurisée les images et documents associés aux propriétés sur IPFS.

Appliquer des règles de cooldown et de limitation de propriétés pour simuler une gestion stratégique.

2. Fonctionnalités Clés

Achat/Vente de propriétés : Les utilisateurs peuvent acheter et vendre des propriétés en utilisant de la crypto-monnaie.

Stockage sur IPFS : Les images et documents des propriétés sont stockés sur IPFS, garantissant une décentralisation et une accessibilité permanente.

Gestion des échanges entre utilisateurs : Possibilité d'échanger des propriétés sous certaines conditions.

CoolDown et Limitation d'Achats : Un système de cooldown empêche l'achat fréquent de propriétés pour équilibrer le jeu.

3. Technologies Utilisées

Blockchain : Smart contracts déployés sur Ethereum (ou un testnet comme Hardhat).

Solidity : Langage de programmation des contrats intelligents.

IPFS : Pour stocker les fichiers décentralisés.

React.js / Vue.js : Interface utilisateur pour interagir avec les contrats.

Ethers.js / Hardhat : Outils pour déployer et tester les smart contracts.

4. Scénario d'Utilisation

Échange d'une propriété : Un utilisateur peut voir la liste des propriétés d'un autre via son adresse et lui proposer un échange avec l'une des siennes.

Mise en vente : Un utilisateur possédant une propriété peut la mettre en vente à un prix défini ainsi que modifier sa surface et sa valeur.

Achat par un autre utilisateur : Un acheteur peut acquérir la propriété via une transaction blockchain.

Application du Cooldown : L'acheteur doit attendre un délai avant de pouvoir acheter une nouvelle propriété.

Vérification sur IPFS : L'image et les documents sont toujours accessibles via l'URL IPFS.

5. Conclusion

Ce projet allie NFTs, blockchain et stockage décentralisé pour offrir une expérience unique d'achat et d'échange de propriétés. L'intégration d'IPFS garantit la sécurité et la pérennité des fichiers associés, tout en maintenant une gestion équilibrée des actifs sur la blockchain.

Choix de Conception et Respect des Contraintes Métiers

1. Choix Technologiques

1.1. Blockchain et Smart Contracts

Ethereum / Hardhat (Testnet) : Utilisation d'un environnement de test pour déployer et valider les contrats.

Solidity : Langage de développement des Smart Contracts assurant la gestion d'achat/vente et échanges de propriétés.

Ethers.js : Interaction entre le frontend et la blockchain.

1.2. Stockage d'Images et Documents

IPFS (InterPlanetary File System) : Permet un stockage décentralisé et immuable des images et documents associés aux propriétés.

1.3. Frontend et Interaction Utilisateur

React.js : Interface utilisateur moderne et performante.

Metamask : Intégration pour la connexion et la gestion des transactions blockchain.

Tailwind CSS : Framework CSS pour un design rapide et adaptatif.

2. Respect des Contraintes Métiers

2.1. Achat/Vente de Propriétés

Les propriétés sont représentées sous forme de NFTs uniques (ERC-721).

Un utilisateur peut acheter une propriété sous certaines conditions (fonds suffisants, propriété en vente, etc.).

Les transactions sont validées sur la blockchain pour assurer transparence et sécurité.

2.2. Règles de Cooldown et Limitations

Cooldown de 5 minutes entre chaque transaction pour empêcher le spam d'achats/ventes.

Limitation à 4 propriétés par utilisateur pour un équilibre de jeu.

2.3. Stockage décentralisé des fichiers

Chaque image/document de propriété est stocké sur IPFS.

Les URLs IPFS sont sauvegardées sur la blockchain pour garantir l'authenticité et l'accessibilité des fichiers.

3. Sécurité et Robustesse

3.1. Protection des Transactions

Require() dans Solidity : Empêche les achats non autorisés et vérifie les fonds.

Signature et validation Metamask : Assure que l'utilisateur valide chaque transaction.

3.2. Tests et Simulations

Hardhat : Tests unitaires et d'intégration sur les smart contracts.

4. Conclusion

Le projet Web3 Monopoly utilise des technologies modernes pour assurer un système d'achat, de vente et d'échange de propriétés décentralisé, sécurisé et performant. L'intégration de IPFS garantit une conservation fiable des images et documents, tout en respectant les contraintes de scalabilité et d'équité du jeu.