



UNIVERSITE ABDELMALEK ESSAADI

Master IA et science de données

Module : BlockChain

Devoir Noté : Twitter DAPP

Réalisé PAR :

Aymane Mahri

Encadré PAR :

Pr. Ikram Ben abdel ouahab

Table des matières :

Introduction :	3
Mise en place :	4
Réalisation :	9
Conclusion :	13

Introduction :

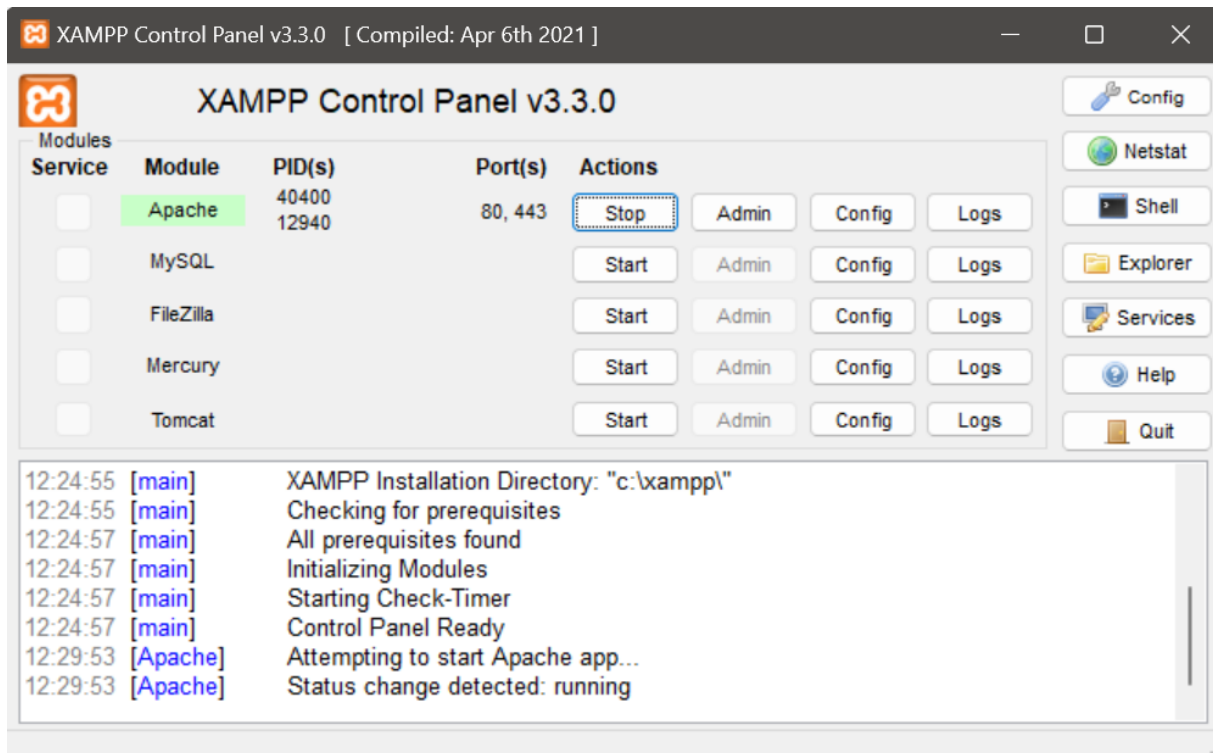
Ce projet vise à développer une interface web permettant aux utilisateurs d'interagir avec le smart contract **MiniSocial** déployé sur la blockchain Ethereum, dans un format de mini réseau social similaire à Twitter. En ajoutant des fonctionnalités telles que l'authentification par portefeuille, la création et la modification de posts, ainsi que la gestion des réactions (likes et dislikes), cette interface démontre le potentiel de la décentralisation pour les réseaux sociaux. Le projet met également en œuvre des vérifications de propriété, permettant uniquement aux auteurs de modifier leurs publications.

Ce rapport détaille les étapes de développement de cette interface, les fonctionnalités implémentées, ainsi que les tests et résultats obtenus.

Mise en place :

Avant de tester notre mini réseau social réalisé, voici les étapes à considérer avant d'y accéder :

Il faudra tout d'abord commencer par Installer Xampp qui nous permettra d'accéder à notre site web via Localhost :



Il faudra aussi avoir déjà installé Node.js et yarn pour ensuite installer Hardhat via la commande suivante : **npm install hardhat**

Après avoir fait tout cela on peut lancer notre hardhat via la commande suivante : **yarn hardhat node**

Ce qui nous donnera accès à 20 comptes avec leurs clés privées :

```

C:\Users\USER\hhproject\chain>yarn hardhat node
yarn run v1.22.2
warning package.json: No license field
$ C:\Users\USER\node_modules\.bin\hardhat node
Started HTTP and WebSocket JSON-RPC server at http://127.0.0.1:8545/

Accounts
=====

WARNING: These accounts, and their private keys, are publicly known.
Any funds sent to them on Mainnet or any other live network WILL BE LOST.

Account #0: 0xf39Fd6e51aad88F6F4ce6aB8827279cFfFb92266 (10000 ETH)
Private Key: 0xac0974bec39a17e36ba4a6b4d238ff944bacb478cbed5efcae784d7bf4f2ff80

Account #1: 0x70997970C51812dc3A010C7d01b50e0d17dc79C8 (10000 ETH)
Private Key: 0x59c6995e998f97a5a0044966f0945389dc9e86dae88c7a8412f4603b6b78690d

Account #2: 0x3C44CdDdB6a900fa2b585dd299e03d12FA4293BC (10000 ETH)
Private Key: 0x5de4111afa1a4b94908f83103eb1f1706367c2e68ca870fc3fb9a804cdab365a

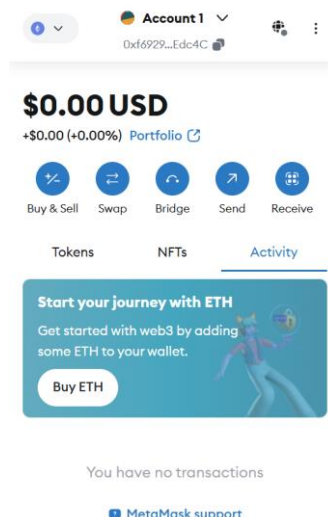
Account #3: 0x90F79bf6EB2c4f870365E785982E1f101E93b906 (10000 ETH)
Private Key: 0x7c852118294e51e653712a81e05800f419141751be58f605c371e15141b007a6

Account #4: 0x15d34AAf54267DB7D7c367839AAf71A00a2C6A65 (10000 ETH)
Private Key: 0x47e179ec197488593b187f80a00eb0da91f1b9d0b13f8733639f19c30a34926a

```

Il faudra laisser le serveur **hardhat** lancé dans le terminal le long de notre test de l'application.

Après cela, il faudra installer **métamask** aussi :



MetaMask est un portefeuille de crypto-monnaies sous forme d'extension de navigateur ou d'application mobile qui permet aux utilisateurs de gérer des comptes Ethereum, d'interagir avec des applications décentralisées (dApps), et d'effectuer des transactions sur la blockchain Ethereum, comme l'envoi et la réception d'Ether (ETH).

Passons maintenant à l'étape suivante, on va créer un nouveau réseau de test dans notre métamask :

[+ Add a custom network](#)

Voici les données à remplir lors de la création de notre nouveau réseau :

The screenshot shows the 'Add a custom network' interface in Metamask. At the top, there's a title bar with a back arrow, the text 'localhost 8545', and a close 'X' button. Below this, the form has several sections: 'Network name' with a text input containing 'localhost 8545', a warning message in orange stating 'According to our records, the network name may not correctly match this chain ID. Suggested name: GoChain Testnet', 'Default RPC URL' with a dropdown menu showing '127.0.0.1:8545', 'Chain ID' with a text input containing '31337', and 'Currency symbol' with a text input containing 'ETH'. Below the currency symbol input, there's another warning message in orange: 'Suggested currency symbol: GO. This token symbol doesn't match the network'. At the bottom of the form is a blue 'Save' button. A vertical scrollbar is visible on the right side of the form.

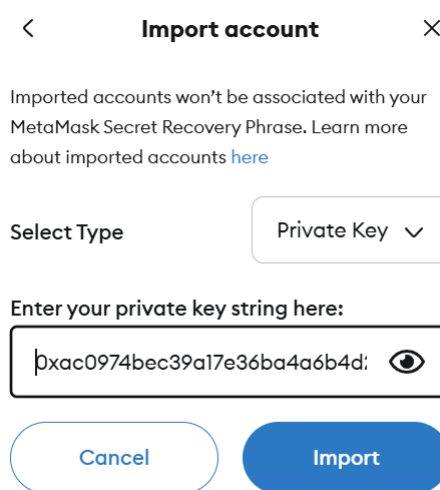
Maintenant nous allons importer 2 des 20 comptes qu'on a obtenu de notre serveur hardhat pour le test :

[+ Add account or hardware wallet](#)

Ensuite :

[📌 Import account](#)

Il suffit maintenant plus que de copier coller les clés privées obtenus des comptes qu'on veut importer :




< **Import account** ×

Imported accounts won't be associated with your MetaMask Secret Recovery Phrase. Learn more about imported accounts [here](#)

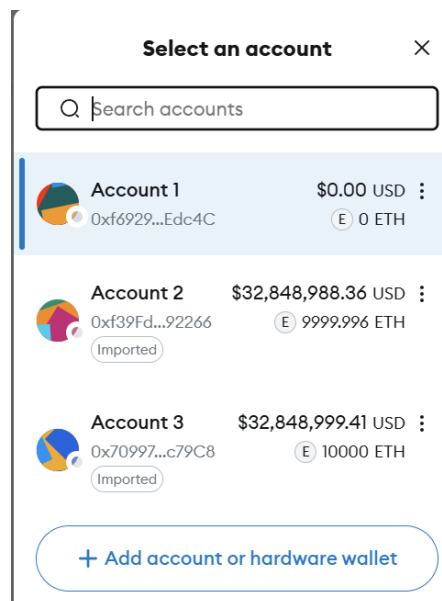
Select Type Private Key ▾

Enter your private key string here:

0xac0974bec39a17e36ba4a6b4d: 




Cancel Import

Après que cela soit fait nous devrions avoir nos compte importer avec notre compte originel au dessus, chaque compte importé possède 10000 ETH :



Select an account ×

Q search accounts

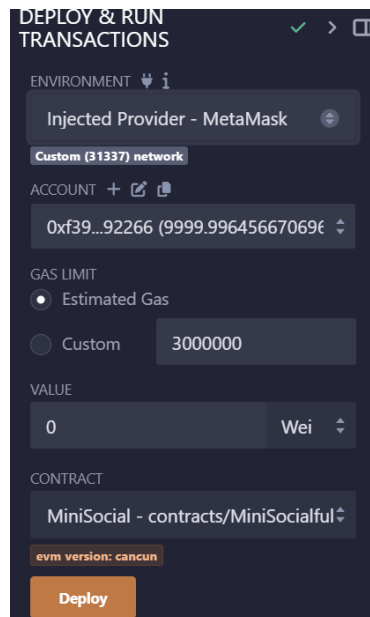
	Account 1 0xf6929...Edc4C	\$0.00 USD : 0 ETH
	Account 2 0xf39Fd...92266 <small>Imported</small>	\$32,848,988.36 USD : 9999.996 ETH
	Account 3 0x70997...c79C8 <small>Imported</small>	\$32,848,999.41 USD : 10000 ETH

+ Add account or hardware wallet

Maintenant, on devra choisir notre réseau de test choisi ainsi que le compte avec lequel on veut tester et c'est tout.

Passons maintenant au site **RemixIDE** pour notre code **Solidify** :

Après avoir fini l'écriture de notre code on devra le compiler comme d'habitude et ensuite le déployer, mais pour ce qui est du déploiement on choisira l'environnement **Injected Provider – MetaMask** ainsi que l'un des comptes importés :

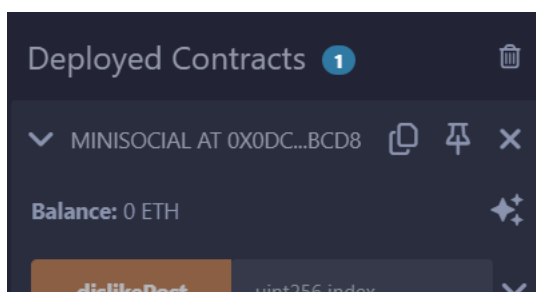


On aura besoin ensuite du **ABI** ainsi que **l'adresse du contract** pour notre code html, voici comment y accéder depuis notre interface **RemixIDE** :

Pour l'ABI dans l'interface Compiler :



Pour l'adresse du contract dans l'interface Deployed Contracts :



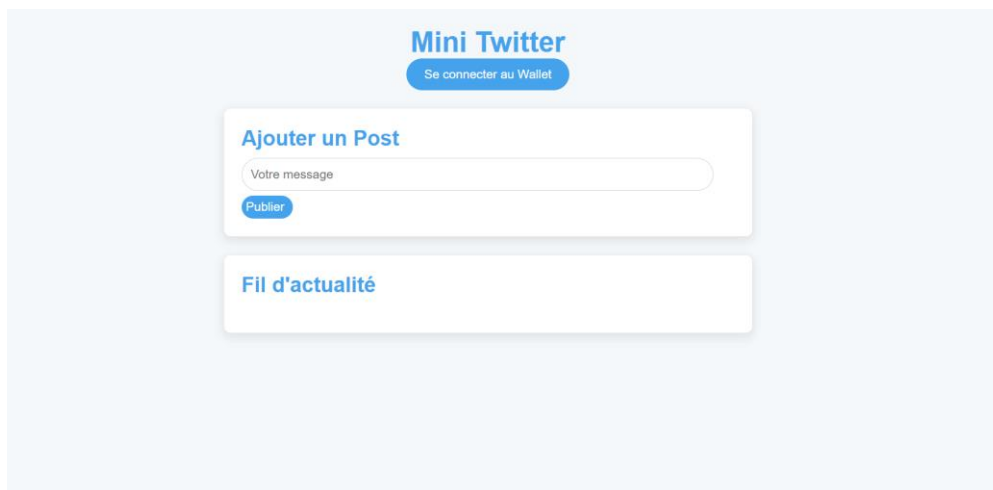
On ajoutera comme script aussi dans notre code html celui de **Web3.js**

Après que tout cela soit mis en place et terminé il suffira d'écrire notre code HTML et de le lancer sur Localhost (Il faut être sûr que apache soit lancé sur XAMPP).

Réalisation :

Après avoir lancé notre code, il est temps d'enfin tester notre Mini réseau social Twitter :

Voici l'interface initiale de notre mini réseau social, il faudra tout d'abord se connecter dans un des comptes importés avant de pouvoir publier un message :



En appuyant sur le bouton Se connecter au Wallet, on est directement connecté au dernier compte choisi lors du déploiement du contract :



On pourra aussi naviguer entre les comptes importés depuis notre mini réseau social en les choisissant depuis notre liste de sélection.

Commençons Maintenant par publier 2 messages dans notre premier compte importé :

Le premier Message :

Mini Twitter

Se connecter au Wallet

0xf39fd6e51aad88f6f4ce6ab8827279cfff92266

Compte connecté : 0xf39fd6e51aad88f6f4ce6ab8827279cfff92266

Ajouter un Post

Premier Message du Compte 1.

Publier

Fil d'actualité

Message: Premier Message du Compte 1.

Auteur: 0xf39Fd6e51aad88F6F4ce6aB8827279cFfB92266

Likes: 0, Dislikes: 0

Date de création: 11/11/2024, 6:26:07 PM

Like

Remove Like

Dislike

Remove Dislike

Modifier

Le Deuxième message :

Se connecter au Wallet

0xf39fd6e51aad88f6f4ce6ab8827279cfff92266

Compte connecté : 0xf39fd6e51aad88f6f4ce6ab8827279cfff92266

Ajouter un Post

Deuxième Message du Compte 1.

Publier

Fil d'actualité

Message: Premier Message du Compte 1.

Auteur: 0xf39Fd6e51aad88F6F4ce6aB8827279cFfB92266

Likes: 0, Dislikes: 0

Date de création: 11/11/2024, 6:26:07 PM

Like

Remove Like

Dislike

Remove Dislike

Modifier

Message: Deuxième Message du Compte 1.

Auteur: 0xf39Fd6e51aad88F6F4ce6aB8827279cFfB92266

Likes: 0, Dislikes: 0

Date de création: 11/11/2024, 6:26:58 PM

Like

Remove Like

Dislike

Remove Dislike

Modifier

Passons Maintenant au deuxième compte et publions un message :

Fil d'actualité

Message: Premier Message du Compte 1.

Auteur: 0xf39Fd6e51aad88F6F4ce6aB8827279cFFb92266

Likes: 0, Dislikes: 0

Date de création: 11/11/2024, 6:26:07 PM

Like

Remove Like

Dislike

Remove Dislike

Message: Deuxième Message du Compte 1.

Auteur: 0xf39Fd6e51aad88F6F4ce6aB8827279cFFb92266

Likes: 0, Dislikes: 0

Date de création: 11/11/2024, 6:26:58 PM

Like

Remove Like

Dislike

Remove Dislike

Message: Premier Message du Compte 2.

Auteur: 0x70997970C51812dc3A010C7d01b50e0d17dc79C8

Likes: 0, Dislikes: 0

Date de création: 11/11/2024, 6:28:24 PM

Like

Remove Like

Dislike

Remove Dislike

Modifier

On peut constater que la publication a été faite avec succès chaque publication est accompagnée du message publié, le compte qui l'a publié, nombre de likes et dislikes, ainsi que la date de création.

On peut aussi remarque aue chaque compte ne peut modifier que les publications qu'il a publié lui-même, essayons de modifier la publication du compte 2 :

Modifier votre message :

Premier Message du Compte 2 modifié.

OK

Cancel

Le changement est Fait ainsi :

Message: Premier Message du Compte 2 modifié.
Auteur: 0x70997970C51812dc3A010C7d01b50e0d17dc79C8
Likes: 0, Dislikes: 0
Date de création: 11/11/2024, 6:28:24 PM
Date de modification: 11/11/2024, 6:33:23 PM

- Like
- Remove Like
- Dislike
- Remove Dislike
- Modifier

On pourra que la date de modification est ajouté lors de la modification de la publication.

Testons maintenant un peut les fonctionnalités de Like et Dislike de notre mini réseau social, chaque Compte a le droit d'une seule opération par publication, soit un Like soit un Dislike pas les deux, pour Disliker il faudra tout d'abord retirer notre Like si on a déjà liker le post précédemment :

Essayons maintenant avec chaque compte que nous possédons de liker chaque publication à lui-même et de disliker les autres, Voici les résultats :

Message: Premier Message du Compte 1.
Auteur: 0xf39Fd6e51aad88F6F4ce6aB8827279cFfB92266
Likes: 1, Dislikes: 1
Date de création: 11/11/2024, 6:26:07 PM

- Like
- Remove Like
- Dislike
- Remove Dislike
- Modifier

Message: Deuxième Message du Compte 1.
Auteur: 0xf39Fd6e51aad88F6F4ce6aB8827279cFfB92266
Likes: 1, Dislikes: 1
Date de création: 11/11/2024, 6:26:58 PM

- Like
- Remove Like
- Dislike
- Remove Dislike
- Modifier

Message: Premier Message du Compte 2 modifié.
Auteur: 0x70997970C51812dc3A010C7d01b50e0d17dc79C8
Likes: 1, Dislikes: 1
Date de création: 11/11/2024, 6:28:24 PM
Date de modification: 11/11/2024, 6:33:23 PM

- Like
- Remove Like
- Dislike
- Remove Dislike

Ainsi nous pouvons conclure que notre mini réseau social marche avec succès, toute les fonctionnalités demandées sont présentes et marche de manière correcte !

Conclusion :

Le développement de cette interface web pour le mini réseau social décentralisé a permis de mettre en place une version simplifiée de Twitter fonctionnant sur la blockchain Ethereum. Toutes les fonctionnalités essentielles sont opérationnelles, et l'interface est intuitive pour l'utilisateur. Ce projet a démontré les avantages de la décentralisation pour les réseaux sociaux, en particulier en ce qui concerne la transparence, la propriété des données et la sécurité des interactions.

Les tests ont montré que le smart contract gère efficacement les publications et réactions des utilisateurs, tandis que la vérification d'auteur permet de garantir la sécurité et la confidentialité des modifications. Ce mini Twitter peut être enrichi de fonctionnalités additionnelles comme des commentaires ou des notifications, et adapté à d'autres plateformes de réseaux sociaux décentralisés.