SORBONNE UNIVERSITÉ

CRÉATEURS DE FUTURS
DEPUIS 1257

Master 2 Informatique
Spécialité STL

UE DAAR

>>> Decentralized Wikipedia

Camilia HEMAIZI
Heythem NOUARI

06 / 12 / 2020

Sommaire

1 Implémentation

Après avoir connecté notre application avec ganache et MetaMask Nous avons rencontré un problème qui nous a pris beaucoup de temps et nous n'avons pas réussi a le réglé jusqu'au dernier jour qui est :

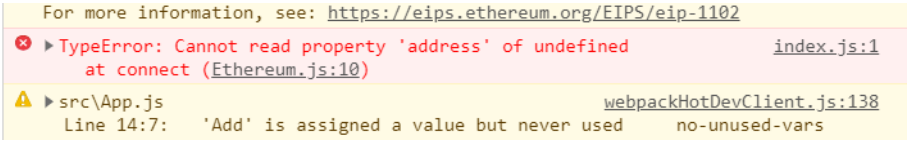

Figure 1. Problème de la non créaton d'une instance Wikipedia

Nous avons quand même essayé de coder sans pouvoir tester chose qui était très difficile car il s'agit aussi de notre premier contact avec ReactJS .

1.1 Contracts

Voici notre Contrat Wikipedia :

```
contract Wikipedia {
 uint public  nbArt = 0 ;
 struct Article {
   uint id;
   string content;
 }

 uint[] public ids;
 mapping (uint => Article) public articlesById;

 constructor() public {
  /* uint index = 0;
   ids.push(index);
   Article memory newArticle = Article(index,"This is your first article in your contract");
   articlesById[index] = newArticle;
   */ addArticle("first Article") ;
 }

 function articleContent(uint index) public view returns (string memory) {
   return articlesById[index].content;
 }

 function getAllIds() public view returns (uint[] memory) {
   return ids;
 }

 function addArticle(string memory cont ) public {
     nbArt ++ ;
     articlesById[nbArt] = Article(nbArt,cont);
 }
 function editArticle(string memory cont , uint id ) public {

     articlesById[id] = Article(id,cont);
 }
```

Figure 2. le contract Wikipedia

Nous avons dans cette classe quatre fonctions :

- **articleContent** : qui nous rend le contenu d'un article en lui passant sans id .
- **getAllIds** :qui nous retourne une liste de tous les ids
- **addArticle** : qui rajoute un article dans notre wikipedia
- **editArticle** : qui modifie le contenu d'un article dans notre wikipedia

Nous pensons que ces quatre fonctions sont suffisantes pour réaliser les fonctionnalités demandées.

1.2 App.js

1.2.1 AllArticles

Nous pensons que ce qu'on a écrit est juste dans ce composant sauf qu'on ne peut pas essayer à cause de l'erreur venant de la classe **Ethureum.js** .

1.2.2 AddArticle

Nous avons mis en commentaire ce que nous avons essayé (car ça ne fonctionne pas) la difficulté était de bien placé la fonction **add** pour pouvoir l'apppeler **OnSubmit** et aussi de pouvoir passer l'input (le contenu du message).

2 Conclusion

Malgrés ne pas pouvoir exécuter l'application a cause de l'erreur dans **Ethereum.js** nous avons essayé de vous montré notre vision de la solution à ce problème. Avec les autres projets et la préparation des éxamens nous n'avons pas pu consacré plus de 3 jours à ce projet. Rajoutant a celà que les deux membres du groupe n'ont jamis utiliser ReactJS chose qui nous a beacoup retardé (et nous pensons que nous devons s y mettre car c'est l'une des technologies les plus demandés en Front).