

Rapport de projet de DAAR

Projet 2 : Decentralized Wikipedia

Julien Simonnet

Alice de Berny

Introduction

Dans ce projet, nous avons tenté de construire un site similaire à Wikipédia, dans lequel des utilisateurs peuvent déposer, lire et modifier des articles, mais avec un backend en « Solidity ». Nous parlerons tout d'abord de la structure de notre backend, avant d'aborder l'architecture du site à proprement, puis nous conclurons en mentionnant nos déconvenues.

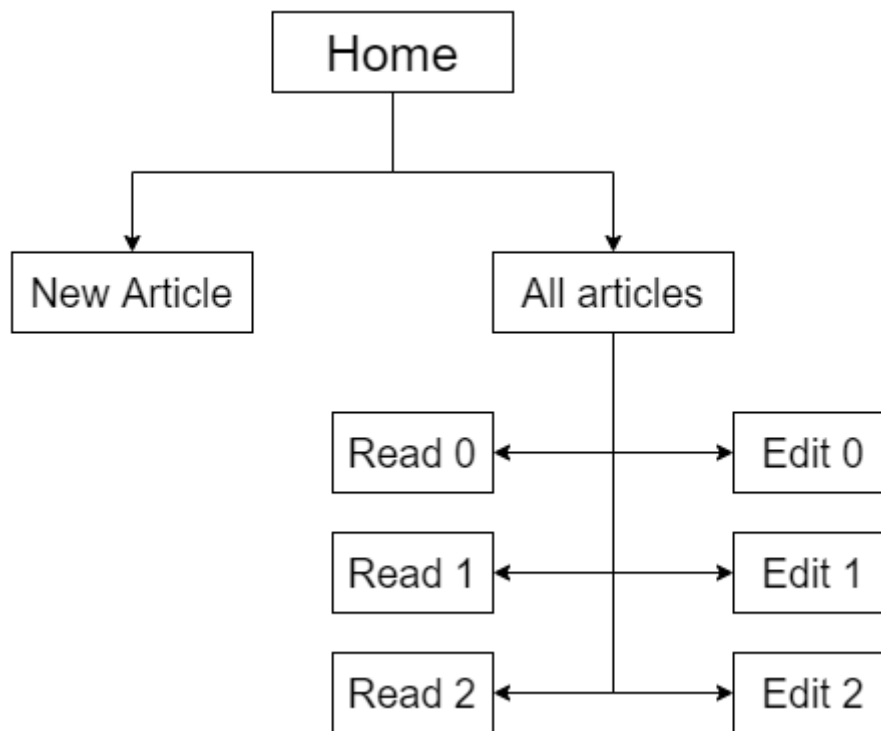
I. Structures de données

Notre backend utilise deux objets : un tableau redimensionnable et une table d'associations. Dans le premier, nous stockons des identifiants et dans le second, nous associons à chaque identifiant un article, un article étant lui-même une structure contenant les informations qui lui sont relatives (comme son contenu). De cette façon, on peut aisément ajouter des propriétés aux articles si cela devait nécessaire.

Cependant, bien que cette organisation permette de facilement accéder aux données, notamment car le tableau d'identifiants est un tableau d'entiers et peut donc être communiquer directement à notre interface Javascript, elle n'est pas sans défaut. En particulier, garder un tableau redimensionnable en (« storage ») le rend beaucoup plus coûteux en « gaz ». De plus, telle qu'elles sont présentés ici, nos structures ne promettent pas de suppression facile, bien au contraire (ce n'est pas gênant ici, mais le devrait si nous voulions enrichir notre système par la suite).

II. Architecture

1. Plan



Ici, nous vous montrons un plan de notre site. Comme vous pouvez le voir, depuis la page d'accueil, on peut se diriger vers la page de création d'un article ou vers une page de consultation, et depuis cette dernière, on peut soit lire, soit modifier un article.

2. Communications

La communication se fait par l'usage de « call » quand il s'agit de lire une information sur la base de données et de « send » quand il s'agit de modifier une information dans celle-ci. Ainsi, qu'il s'agisse d'afficher le contenu d'un article ou donner la liste des articles présents, un « call » sera utilisé. En revanche, quand il faut ajouter ou modifier un article, le texte est envoyé sur la blockchain par une transaction grâce à un « send ».

III. Problèmes rencontrés

Nous avons malheureusement rencontré divers problèmes durant ce projet, certains que nous ne sommes d'ailleurs pas parvenu à résoudre. L'un d'entre eux est que, du moins sur nos machines, l'ajout ou la modification d'un article par une transaction sur la blockchain nécessite souvent plusieurs tentatives et ce bien que toutes les valeurs utilisées dans l'opération soient à première vue

valides. Nous avons d'abord pensé que nous manquions de gaz, mais fixer la quantité de gaz à utiliser n'a pas résolu le problème.

Un autre problème que nous avons rencontré est que l'ajout d'un historique de modifications semble nécessiter deux choses : un moyen de comparer en détails deux strings, que soit en Javascript ou en Solidity, et un moyen de sauvegarder des informations supplémentaires dans nos articles. Dans une première approche, nous avons pensé à stocker un tableau de modifications dans chaque article, mais avons été confronté à un problème : il n'est pas possible de créer un tableau redimensionnable en mémoire (« memory »), soit il est de taille fixe ou soit il est gardé en « stockage », ce qui coûterait très cher, particulièrement pour chaque article. C'est pourquoi, de peur qu'augmenter drastiquement l'usage de gaz n'affecte négativement le reste de notre système, nous avons renoncé à traiter le problème.

Conclusion

Au cours de ce projet, nous avons découvert que certains systèmes de transactions par blockchain peuvent être utilisés comme des bases de données et bien que je n'aie pas personnellement apprécié le codage d'un frontend, j'admettrais que c'était assez enrichissant. Enfin, je dirais que, compte tenu de la récente popularité de la technologie blockchain, et bien que je préférasse les bases de données classiques, ce travail pourrait bien s'avérer profitable si nous venions à rencontrer de la décentralisation ou de la communication par transactions.