MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE ABDELHAMID IBN BADIS - MOSTAGANEM

Faculté des Sciences Exactes et d'Informatique

Département de Mathématiques et informatique

Filière : Informatique

MEMOIRE DE FIN D'ETUDES

Pour l'Obtention du Diplôme de Master en Informatique

Option : **Ingénierie des Systèmes d'Information**

Présenté par :

BENALI Feriel ET GUENOUNA Abdelkader

THEME :

Les questions de confidentialité dans les organisations décentralisées sur la blockchain

Soutenu le :

Devant le jury composé de :

HOCINE Nadia MCA Université de Mostaganem Président

BENTAOUZA Chahinez MCB Université de Mostaganem Examinateur

MIROUD Mohammed El Mustapha MAA Université de Mostaganem Encadreur

Année Universitaire 2023-2024

Résumé

L’objectif de notre projet de fin d’étude est d'aborder la problématique de la confidentialité au sein des organisations décentralisées autonomes (DAO). Pour ce faire, nous avons étudié le fonctionnement de la blockchain, une base de données distribuée qui enregistre les transactions de manière transparente et sécurisée grâce à des algorithmes cryptographiques complexes.

Nous avons analysé la structure de la blockchain et ses composants de base, ainsi que les fondements cryptographiques qui garantissent l’intégrité et la sécurité des informations.

Ensuite, nous avons exploré le concept des DAO, qui opèrent de manière autonome grâce à des smart contracts (contrats intelligents) exécutés sur la blockchain. Les DAO sont conçues pour fonctionner sans gestion centralisée, en utilisant des règles prédéfinies codées dans des smart contracts pour régir leurs opérations et prises de décisions.

La décentralisation offre plusieurs avantages, notamment la transparence, l'ummutabilité et la sécurité.

Dans le cadre de ce mémoire, nous avons également élaboré un protocole destiné à résoudre les problèmes de transparence financière des DAO tout en intégrant des fonctionnalités de confidentialité.

.

Mots-clés:

Blockchain, DAOs, (organisations décentralisées autonomes), bitcoin, ethereum, cryptographie, contrat intelligent, Consensus. SNARK, zero knowledge proofs, tornado cash. anonymat

Abstract

The aim of our end-of-studies project is to address the issue of privacy within Decentralized Autonomous Organizations (DAOs). To do this, we studied the operation of blockchain, a distributed database that records transactions transparently and securely thanks to complex cryptographic algorithms.
We analyzed the structure of the blockchain and its basic components, as well as the cryptographic foundations that guarantee the integrity and security of information.
Next, we explored the concept of DAOs, which operate autonomously thanks to smart contracts executed on the blockchain. DAOs are designed to operate without centralized management, using predefined rules coded in smart contracts to govern their operations and decision-making.
This decentralization offers several advantages, including transparency, immutability, and security.
As part of this thesis, we also developed a protocol aimed at solving the problems of financial transparency of DAOs while integrating privacy features.

Keywords:

Blockchain , Decentralized Autonomous Organizations, DAOs ,Bitcoin ,Ethereum ,Cryptography ,Smart contract, Consensus, SNARK, Zero knowledge proofs, anonymity, tornado cash.

ملخص

يهدف مشروع تخرجنا إلى معالجة مشكلة الخصوصية داخل المنظمات المستقلة
ولتحقيق ذلك، قمنا بدراسة كيفية عمل تقنية البلوكشين، وهي قاعدة بيانات موزعة تسجل المعاملات بطريقة شفافة .وآمنة بفضل خوارزميات التشفير المعقدة
.قمنا بتحليل بنية البلوكشين ومكوناتها الأساسية، بالإضافة إلى الأسس التشفيرية التي تضمن سلامة وأمان المعلومات
، التي تعمل بشكل مستقل بفضل العقود الذكية التي تُنفذ على (DAO) بعد ذلك، استكشفنا مفهوم المنظمات المستقلة البلوكشين. تم تصميم المنظمات المستقلة للعمل دون إدارة مركزية، باستخدام قواعد محددة مسبقًا مشفرة في العقود .الذكية لتنظيم عملياتها واتخاذ القرارات
تقدم هذه اللامركزية عدة مزايا، من بينها الشفافية، والثبات، والأمان
في إطار هذا البحث، قمنا أيضًا بتطوير بروتوكول يهدف إلى حل مشاكل الشفافية المالية في المنظمات المستقلة مع دمج ميزات الخصوصية

كلمات مفتاحية

البلوكشين، المنظمات المستقلة، البيتكوين، الإثريوم، التشفير، العقد الذكي، الإجماع السرية كاش ، تورنادو ،

SNARK دلائل الصفرية المعرفة.

Dédicaces

Je dédie ce travail précieux à mes chers parents,qui ont toujours été ma source d'inspiration dont leur soutien indéfectible et les sacrifices ont façonné mon chemin. Votre présence et vos encouragements constants ont été la lumière qui a éclairé mon parcours et à toute ma famille.

Je le dédie également ce travail à
tous les enseignants qui ont partagé leurs connaissances et leur expertise et qui m'ont aidé dans mon parcours académique. Votre contribution a été précieuse pour mon développement.

À tous ceux qui ont été des soutiens essentiels à ma réussite, que ce soit par leurs encouragements, leurs conseils ou leurs actions, je vous adresse toute ma reconnaissance.

Feriel Benali

Je dédie ce travail :
À ma famille, pour leur amour et leur soutien continus.
À mes amis, pour leur encouragement et leur compréhension.
À mon directeur de mémoire, pour sa patience et ses précieux conseils.
À mes collègues et collaborateurs, pour leur aide et leur coopération tout au long de cette aventure.

Guenouna Abdelkader

Remerciements

Nous exprimons notre gratitude envers Dieu Tout Puissant pour nous avoir accordé la santé et la détermination nécessaires pour entreprendre et achever ce mémoire. Nous tenons également à remercier chaleureusement toutes les personnes qui nous ont apporté un soutien moral et technique, contribuant ainsi à rendre notre environnement de travail agréable.

Nous souhaitons exprimer notre profonde reconnaissance envers Monsieur MIROUD Mohammed El Mustapha pour son encadrement exceptionnel, sa patience, sa rigueur et sa disponibilité tout au long de la préparation de ce travail.

Nous tenons également à remercier l'ensemble de nos professeurs pour leur générosité et leur patience.

Nous tenons également à remercier chaleureusement les étudiants du Master 2 Informatique ISI pour leur soutien moral et leur attitude positive tout au long de notre parcours scolaire.

Enfin nous souhaitons exprimer notre gratitude à toute l'équipe de la Faculté des Sciences Exactes et de l'Informatique (FSEI) de Mostaganem, ainsi qu'à tous les enseignants et membres du personnel, pour leur soutien et leur dévouement.

Liste des figures

Liste des abréviations

Abréviation	Expression Complète	Page
SHA-256	Secure Hash Algorithm 256-bit	9
DAO	Organisation Aécentralisé Autonome	23
DAC	Corporation Organisation Décentralisé	27
NFT	Non-Fungible Token	43
PDAO	Protocole Organisation Oécentralisé Autonome	55
DApp	Application Décentralisée	74
CLI	Interface en ligne de commande	74
SGBDR	Système de Gestion de Base de Données Relationnelle	76
GPL	Licence Publique Générale	76
MIT	Massachusetts Institute of Technology	77

Table des matières

Introduction Générale

Les blockchains représentent une innovation technologique majeure qui transforme de nombreux aspects de notre société, allant de la gestion des systèmes d'information à la finance, en passant par les échanges commerciaux. Apparue pour la première fois avec la publication du livre blanc du Bitcoin par le mystérieux Satoshi Nakamoto en 2008, la blockchain a rapidement attiré l'attention en raison de son potentiel à offrir une alternative décentralisée aux systèmes centralisés traditionnels.

Cette technologie repose sur des principes cryptographiques solides pour assurer la sécurité et l'intégrité des transactions et des données. Contrairement aux méthodes traditionnelles de sécurisation de l'information, la blockchain utilise un réseau distribué de nœuds pour valider et enregistrer les transactions de manière immuable, rendant la manipulation ou la falsification des données pratiquement impossible. Cela a conduit à une adoption rapide de la blockchain dans des domaines variés, y compris les cryptomonnaies, où elle permet des transactions financières sécurisées sans nécessiter de tiers de confiance.

Parmi les développements les plus récents et les plus prometteurs figurent les organisations autonomes décentralisées (DAO). Ces entités utilisent les principes de la blockchain pour permettre une gouvernance collective et transparente, où les décisions sont prises de manière démocratique par les participants. Les DAO éliminent la nécessité d'intermédiaires, réduisent les barrières à l'entrée, et augmentent la transparence, ce qui les rend attrayantes pour de nombreuses entreprises et communautés cherchant à innover dans leur gestion et leur gouvernance.

Dans ce document, nous étudierons l'impact des blockchains et des organisations autonomes décentralisées (DAO) sur la gouvernance, la sécurité et la confidentialité. Notre exploration se déroulera en quatre chapitres détaillés.

Lors du premier chapitre, nous nous pencherons sur les blockchains, en expliquant leurs bases théoriques, les concepts fondamentaux de la cryptographie et les composants essentiels de cette technologie. Ensuite nous aborderons également les diverses applications de la blockchain dans différents domaines, tels que la santé, les Supply chain et bien d'autres.

Dans le deuxième chapitre, nous approfondirons notre analyse des DAO. Nous examinerons les processus de prise de décision. Nous discuterons également des défis et des problèmes de gouvernance et de sécurité auxquels ces organisations sont confrontées. De plus, nous explorerons les techniques de confidentialité utilisées dans les blockchains, telles que le mixing(mixeur) pour garantir la confidentialité des informations des utilisateurs et les preuves à divulgation nulle de connaissance (Zero Knowledge Proofs). Cette technologie puissante permet de prouver la connaissance d'une information sans la divulguer, offrant ainsi une confidentialité renforcée.

Dans le troisième chapitre, nous nous concentrerons sur un exemple concret : la ConstitutionDAO. Cette étude de cas nous permettra d'examiner les problèmes de confidentialité spécifiques aux DAO. Nous décrirons une technique de confidentialité, le protocole Tornado Cash, un mixeur qui garantit l'anonymat des transactions des utilisateurs. Cette solution a été utilisée pour aborder les problèmes de confidentialité dans notre étude.

La problématique que nous avons traitée dans ce mémoire est la suivante : comment garantir la confidentialité des dépôts dans un DAO, alors que par définition, les DAO sont des organisations transparentes.

Nous proposerons au cours du troisième chapitre notre solution, basée sur tornado cash afin de garantir la confidentialité des dépôts et en même temps donner la possibilité à un contributeur de se rétracter et de recouvrir ses fonds. Le protocole que nous avons proposé est

constitué de trois étapes : La création du DAO, le dépôt de fonds et le retrait de ces fonds. Chaque étape sera décrite en profondeur, en mettant en évidence les mesures prises pour garantir la confidentialité des informations, protéger les fonds et permettre une gestion privée des actifs par les gestionnaires du DAO.

Chapitre 1

La technologie blockchain

1.1 Introduction

La technologie blockchain est une grande base de données de stockage et de transmission d'informations de manière sécurisée et transparente, contrairement à un système centralisé où les données sont stockées sur un serveur unique. Dans ce chapitre, nous définirons la blockchain et expliquerons ses concepts principaux de base, tels que les transactions, les blocs et les nœuds. Ensuite, nous aborderons les mécanismes de sécurité fondamentaux de la blockchain, notamment les fonctions de hachage cryptographique, les structures de données basées sur ces fonctions et les signatures numériques.

Nous conclurons ce chapitre introductif en explorant les différentes applications de la blockchain dans des domaines tels que la santé, la chaîne d'approvisionnement, les organisations autonomes décentralisées (DAO), l'industrie énergétique , les Stock Market et le management d'identité .

.

1.2 Définition :

Les blockchains sont des registres numériques distribués et résistants à la falsification, utilisés sans référentiel central et généralement sans autorité centrale. Ils permettent à une communauté d'utilisateurs d'enregistrer des transactions dans un registre partagé au sein de cette communauté, de sorte que, sous un fonctionnement normal du réseau blockchain, aucune transaction ne puisse être modifiée une fois publiée. Cette technologie, combinée à d'autres technologies et concepts informatiques, a permis de créer des cryptomonnaies

modernes comme Bitcoin, ainsi que d'autres applications. La blockchain enregistre les transactions publiquement et est maintenue par un groupe de participants distribués, ce qui la rend résiliente aux tentatives de modification ultérieure. Elle est composée de nombreux éléments complexes, mais chaque élément peut être compris individuellement comme un élément de base du système global [1].

1.3 Les bases de sécurité de la blockchain:

La sécurité de blockchain se concentre sur la cryptographie notamment les fonctions de hachage cryptographiques pour protéger les données stockées, autrement dit il associé à chaque bloc une empreinte unique pour sécuriser les transactions garantissant la fiabilité , et la confiance entre ses utilisateurs

1.3.1 Les fonctions de hachage cryptographique : C'est une fonction mathématique qui possède les propriétés suivantes :

- Elle peut prendre comme entrée n'importe quel string de n'importe quelle taille.
- La taille de la sortie est fixe, elle peut être par exemple de 256 bits pour la fonction de hachage utilisée par bitcoin.
- Elle est efficacement calculable, cela signifie que le calcule d'une empreinte se fait extrêmement rapidement.

En plus des propriétés que nous avons citées précédemment, les fonctions de hachage cryptographiques possèdent les propriétés suivantes:

- La résistance aux collisions celà veut dire nous ne pouvons trouver deux entrées différentes qui donnent en sortie la même empreinte.
- La résistance à la préimage, cette propriété signifie qu'à partir d'une empreinte, nous ne pouvons pas deviner l'entrée qui la générée (c'est une fonction à sens unique).

- Elles sont Puzzle-freindly (elles permettent de créer des puzzle cryptographique qui sont à la base du minage. Le minage sert à la validation des blocs dans la blockchain et nous en reparlerons dans la suite du document.

Exemple d'empreinte pour SHA-256 (Secure Hash Algorithm 256-bit**) :**

L'empreinte du mot (guenouna) est la suivante :

4e5a0870cc796af9b967aee8f7d3c36abb727d73622d0358efaecb3549949b34

En changeant le "g" minuscule au début du nom en majuscule, nous obtenons une empreinte totalement différente :

l'empreinte(Guenouna): :

822cade9d741143a77b7c1e24120d880e63b7bbd5e762de6fb718782ac259518

Nous remarquons que le moindre changement dans l'entrée de la fonction de hachage produit une sortie totalement différente. Ceci est une propriété importante des fonctions de hachage. Elle permet de détecter les modifications dans les fichiers : Si l'empreinte a changée, donc le fichier est obligatoirement différent.

Propriété 1: Résistance aux collisions :

Une fonction de hachage H est dite résistante aux collisions s'il est pratiquement impossible à trouver deux valeurs, x et y, telles que $x \neq y$, mais $H(x) = H(y)$

Propriété 2 : Résistance à la préimage :

La fonction de hachage H est résistante à la préimage si :

Quand une clé K est choisie parmi un espace de probabilité avec une entropie minimale élevée, si nous connaissons le résultat de H(x|| k) = y , il est infaisable de trouver x sachant y.

En d'autres termes, la fonction est à sens unique.

Propriété 3 :Puzzle freindly :

Une fonction de hachage H est dite "Puzzle freindly" si, pour chaque valeur de sortie possible "y" de "n" bits, si k est choisi à partir d'une distribution avec une entropie minimale élevée, alors il est difficile de trouver x tel que H(k || x) = y en un temps significativement inférieur à 2^n.

Intuitivement, cela signifie que si quelqu'un veut cibler la fonction de hachage pour obtenir une valeur de sortie particulière, et qu'il y a une partie de l'entrée qui est choisie de manière suffisamment aléatoire, il est très difficile de trouver une autre valeur qui atteint exactement cette cible

1.3.1.2 Les structures de données basées sur les fonctions de hachage (arbre de Merkle, blockchain):

Hash pointer:

C'est un pointeur qui pointe vers l'endroit où une information est stockée accompagné d'un hachage cryptographique des informations. Les hash pointers servent à récupérer l'information et à vérifier qu'elle n'a pas été modifiée comme montré dans la figure 1. ci-dessous.

Figure 1 :Hash pointer

Structure de blockchain par hash pointer :

Dans la Figure 2 nous avons construit une liste chaînée en utilisant des pointeurs de hachage. Nous allons appeler cette structure de données une chaîne de blocs **(blockchain)** . Alors que dans une liste chaînée classique, vous avez une série de blocs, chaque bloc contenant des données ainsi qu'un pointeur vers le bloc précédent dans la liste, dans une chaîne de blocs, le pointeur vers le bloc précédent sera remplacé par un pointeur de hachage. Ainsi, chaque bloc nous indique non seulement où se trouvait la valeur du bloc précédent, mais il contient également un condensé de cette valeur qui nous permet de vérifier que la valeur n'a pas changé. Nous stockons la tête de la liste, qui est simplement un pointeur de hachage régulier pointant vers le bloc de données le plus récent.

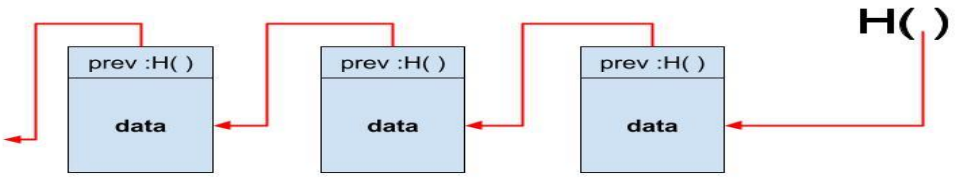

Figure 2 :La Structure de blockchain

Détection de la modification :

Un cas d'utilisation d'une chaîne de blocs est un journal inviolable. C'est-à-dire que nous voulons construire une structure de données de journal qui stocke un ensemble de données et nous permet d'ajouter des données à la fin du journal. Mais si quelqu'un modifie des données qui se trouvent plus tôt dans le journal, nous allons le détecter.

Pour comprendre pourquoi une chaîne de blocs atteint cette propriété inviolable, demandons-nous ce qui se passe si un adversaire veut altérer des données qui se trouvent au milieu de la chaîne. Plus précisément, l'objectif de l'adversaire est de le faire de manière à ce que quelqu'un qui se souvient uniquement du pointeur de hachage à la tête de la chaîne de blocs ne puisse pas détecter la manipulation. Pour atteindre cet objectif

- L'adversaire change les données d'un bloc k. Puisque les données ont été modifiées, le hachage dans le bloc k + 1, qui est un hachage de l'ensemble du

bloc k, ne correspondra pas. Sachant que le nouveau hachage ne correspondra pas au contenu modifié, car la fonction de hachage est résistante aux collisions. Ainsi, nous détectons l'incohérence entre les nouvelles données dans le bloc k et le pointeur de hachage dans le bloc k + 1. Comme montré dans la figure 3.

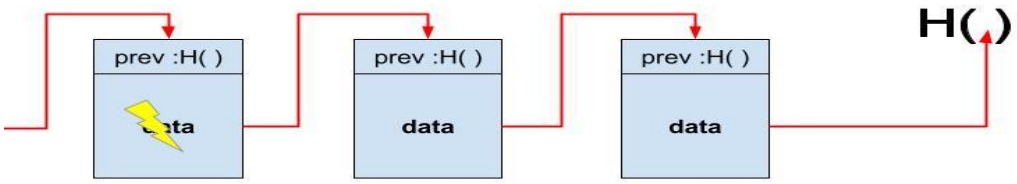

Figure 3 : Le cas de modifier le bloc k.

- L'adversaire peut continuer à essayer de dissimuler cette modification en changeant également le hachage du bloc suivant. L'adversaire peut continuer à faire cela, mais cette stratégie échouera lorsqu'il atteindra la tête de la liste.voir la figure 4.

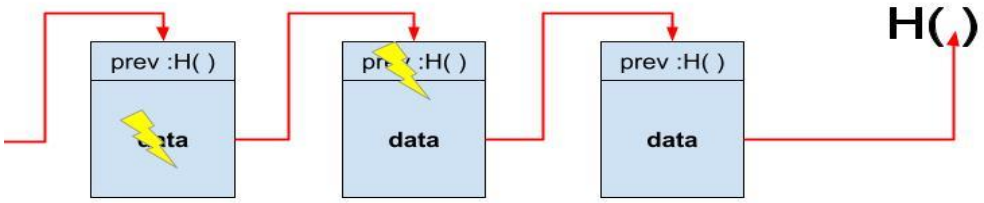

Figure 4 : Le cas de modifier le bloc suivant(k+1).

- En particulier, tant que nous stockons le pointeur de hachage de la tête de la liste dans un endroit où l'adversaire ne peut pas le modifier, l'adversaire ne pourra pas modifier les blocs sans être détecté. Voir la figure 5.

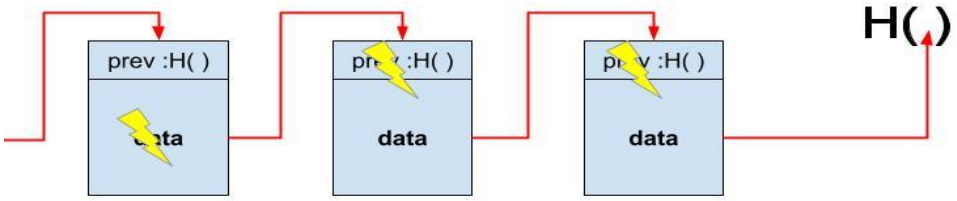

Figure 5. : Le cas d'atteindre la tête de la liste .

Les Arbres de Merkle :

Une autre structure de données utile que nous pouvons construire en utilisant des pointeurs de hachage est un arbre binaire. Un arbre binaire avec des pointeurs de hachage est appelé un arbre de Merkle, du nom de son inventeur Ralph Merkle. Supposons que nous ayons un certain nombre de blocs contenant des données. Ces blocs constituent les feuilles de notre arbre. Nous regroupons ces blocs de données par paires, et pour chaque paire, nous construisons une structure de données qui a deux pointeurs de hachage, un vers chacun de ces blocs. Ces structures de données constituent le niveau supérieur de l'arbre. À leur tour, nous regroupons ces structures en groupes de deux, et pour chaque paire, nous créons une nouvelle structure de données qui contient le hachage de chacun. Nous continuons à faire cela jusqu'à ce que nous atteignons un seul bloc, la racine de l'arbre. La figure 6 représente la structure Arbre de Merkle

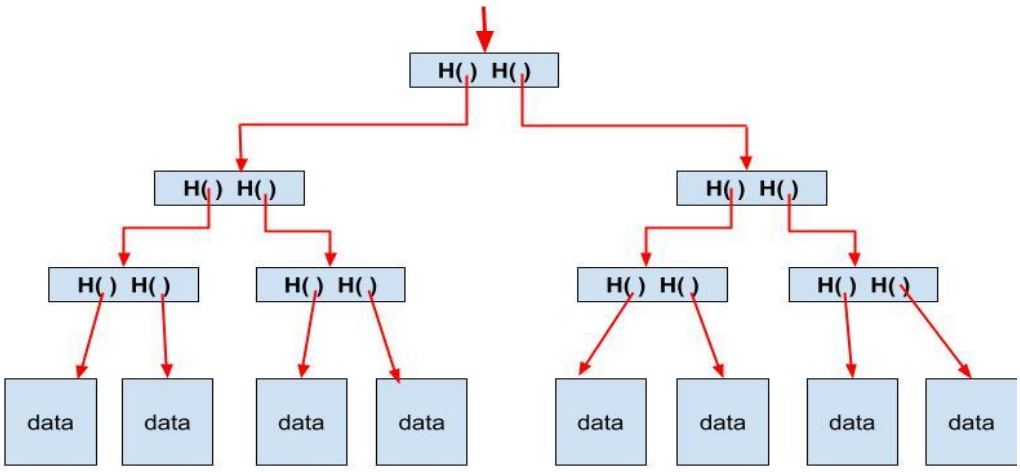

Figure 6: La structure d'Arbre de Merkle

Dans un arbre de Merkle, les blocs de données sont regroupés par paires et le hachage de chaque bloc est stocké dans un nœud parent. Les nœuds parents sont à leur tour regroupés par paires et leurs hachages sont stockés un niveau plus haut dans l'arbre. Ce processus continue jusqu'à ce que nous atteignons le nœud racine.

Comme précédemment, nous ne retenons que le pointeur de hachage à la tête de l'arbre. Nous avons maintenant la capacité de traverser les pointeurs de hachage vers le bas pour atteindre n'importe quel point dans la liste. Cela nous permet de nous assurer que les données n'ont pas été altérées car, tout comme avec la blockchain, si un adversaire altère un bloc de données en bas de l'arbre, cela entraînera le pointeur de hachage du niveau supérieur à ne pas correspondre, et même s'il continue d'altérer ce

bloc, le changement se propagera finalement jusqu'au sommet de l'arbre où il ne pourra pas altérer le pointeur de hachage que nous avons stocké. Ainsi, toute tentative de manipulation de données sera détectée en se contentant de retenir le pointeur de hachage en haut.

Preuve d'appartenance:

Une autre caractéristique intéressante des arbres de Merkle est que, contrairement à la blockchain que nous avons construite précédemment, elle permet une preuve concise d'appartenance. Disons que quelqu'un veut prouver qu'un certain bloc de données est un membre de l'arbre de Merkle. Comme d'habitude, nous ne retenons que la racine. Ensuite, ils doivent nous montrer ce bloc de données et les blocs sur le chemin du bloc de données jusqu'à la racine. Nous pouvons ignorer le reste de l'arbre, car les blocs sur ce chemin suffisent pour nous permettre de vérifier les hachages jusqu'à la racine de l'arbre,voir la figure 7.

Si l'arbre contient “n” nœuds, seuls environ log(n) éléments doivent être montrés. Et puisque chaque étape ne nécessite que le calcul du hachage du bloc enfant, il faut environ log(n) temps pour que nous le vérifiions. Ainsi, même si l'arbre de Merkle contient un très grand nombre de blocs, nous pouvons toujours prouver l'appartenance en un temps relativement court. La vérification s'effectue donc en un temps et un espace logarithmiques par rapport au nombre de nœuds dans l’arbre.

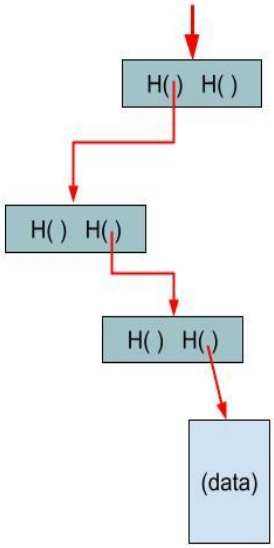

Figure 7 : Preuve d'appartenance.

.

Pour prouver qu'un bloc de données est inclus dans l'arbre, il suffit de montrer les blocs sur le chemin de ce bloc de données jusqu'à la racine. Nous montrons dans le chapitre 3 une utilisation des arbres de Merkle afin de prouver l'appartenance d'un bloc à un arbre.

1.3.2 Les signatures numériques :

Les signatures numériques sont un élément clé de la cryptographie utilisées pour authentifier et garantir l'intégrité des données numériques. Elles fonctionnent comme des signatures

manuscrites mais elles offrent des niveaux de sécurité et de complexité supérieurs à ceux des signatures manuscrites traditionnelles

La signature numérique que nous allons utiliser possède deux propriétés :

- La signature ne peut être créée que par vous, mais n'importe qui peut vérifier sa validité. : une signature numérique créée avec une clé privée peut être vérifiée comme étant valide en utilisant la clé publique correspondante, garantissant ainsi l'authenticité du message signé.

- La signature est liée à un document spécifique pour éviter toute utilisation frauduleuse: Il doit être impossible de falsifier des signatures. Même si un adversaire connaît la clé publique et voit les signatures sur d'autres messages, il ne peut pas falsifier une signature sur un message sans en avoir vu la signature préalablement [2].

1.4 Concepts principaux de la blockchain :

1.4.1 Transaction :

Une transaction dans une blockchain représente une interaction entre les parties. Par exemple, dans le cas des cryptomonnaies, une transaction représente un transfert de la cryptomonnaie entre les utilisateurs du réseau blockchain. Pour les scénarios interentreprises, une transaction peut être un moyen d'enregistrer des activités se produisant sur des actifs numériques ou physiques. Chaque bloc dans une blockchain peut contenir zéro ou plusieurs transactions. Un utilisateur du réseau blockchain envoie des informations au réseau , telles que l'adresse de l'expéditeur (ou un autre identifiant pertinent), la clé publique de l'expéditeur, une signature

numérique, les inputs de la transaction et les outputs de la transaction. Une transaction de cryptomonnaie typique nécessite au moins les informations suivantes :

- Les inputs (une liste des actifs numériques à transférer)
- Les outputs (les comptes qui recevront les actifs numériques et la quantité)
- La preuve que l'expéditeur a accès aux inputs référencés en signant numériquement la transaction[1].

1.4.2 Les blocs :

Les utilisateurs soumettent des transactions au réseau blockchain via des logiciels, qui les envoient à des nœuds du réseau. Les transactions sont ensuite propagées aux autres nœuds mais ne sont pas ajoutées automatiquement à la blockchain. Une transaction en attente doit être ajoutée à un bloc par un nœud émetteur. Les blocs contiennent des en-têtes avec des métadonnées et des données contenant des transactions validées. La validité est assurée en vérifiant que les transactions sont correctement formatées et que les fournisseurs d'actifs numériques ont signé la transaction. Les autres nœuds vérifient les transactions et refusent les blocs avec des transactions invalides.[1]**.**

1.4.3 Les noeuds :

D'un point de vue technique, les membres de la Blockchain sont des ressources informatiques (par ex. des ordinateurs) qui sont préalablement connectées à la Blockchain, suite à une phase d'enrôlement. Ces ressources sont couramment appelées "nœuds" du fait qu'elles sont mises en réseau au travers d'Internet. [3] .

1.4.4 Protocole de Consensus :

Les algorithmes de consensus sont les mécanismes par lesquels les nœuds du réseau blockchain parviennent à un accord sur la validité et l'authenticité des transactions ou des blocs de données. Comme le grand livre des transactions de la blockchain est décentralisé, le

mécanisme de consensus est le processus central qui vérifie et sécurise les blocs de transactions en faisant deux choses. Tout d'abord, l'algorithme de consensus garantit que le prochain bloc ajouté est la seule et unique version de la vérité. Ensuite, l'algorithme empêche tout adversaire de falsifier la chaîne avec succès.
Des exemples de mécanismes de consensus utilisés sur les blockchains sont : proof-of work[4], proof-of-stake [5], delegated proof-of-stake[6], proof-of importance[7], directed acyclic graph[8], and practical Byzantine fault tolerance [9] .

1.5. Contrat intelligent(Smart Contract) :

Les contrats intelligents sont des programmes informatiques autonomes auto-exécutables qui sont exécutés en fonction d'une condition définie par le programmeur . Ces contrats sont capables de faciliter, de faire respecter et d'exécuter des accords entre deux parties en utilisant la blockchain. Les contrats intelligents ont diverses applications possibles telles que :

- Gestion d'entreprise : les entreprises peuvent bénéficier des contrats intelligents et économiser beaucoup de temps et d'argent, ils peuvent établir un contrat intelligent simplement indiquant quand la date est "telle date", les salaires seront envoyés automatiquement aux employeurs
- Paiement : par exemple, on peut payer le loyer de la chambre automatiquement à la fin du mois sans impliquer une banque entre les deux, un développeur écrit un programme informatique system (contrat intelligent). Ce programme définit l'intégralité des règles telles qu'elles ont été définies au début du projet : un mois de souscription, à qui les fonds seront envoyés, quel montant minimum sera récolté, quand les conditions (règles) les conditions sont remplies, telles que la date de paiement, le code sera exécuté et le paiement est effectué automatiquement [10].

1.6. Validation des transactions :

1.6.1. Le minage (mining) :

Le minage dans les réseaux blockchain est une opération où les mineurs utilisent des ordinateurs puissants pour résoudre des problèmes mathématiques complexes liés à chaque transaction. Cela permet de valider et d'ajouter un nouveau bloc à la blockchain environ toutes les 10 minutes. Le minage assure la difficulté de trouver la solution du bloc, assurant ainsi la fiabilité des données. Il s'agit d'un processus sans permission visant à assurer la cohérence mondiale d'un registre décentralisé. Le minage nécessite une consommation énergétique importante et est incité par une récompense en nouvelles pièces ou en frais de transaction[11]. Nous n'avons pas donné de détails sur le processus de minage afin de garder un document relativement court. Pour des informations plus techniques sur le processus de minage, le lecteur peut consulter la référence [2]**.**

1.7 Les Applications de Blockchain dans d'autres domaine que les cryptomonnaies :

Bien que développée à la base dans le but d'être une plateforme de cryptomonnaie, les chercheurs se sont rapidement rendu compte que l'on pouvait lui trouver des utilisations de nombreux autres domaines.

Le domaine de la santé (HealthCare) :

La technologie des registres distribués, notamment la blockchain, offre des solutions innovantes pour transformer les services de santé. Elle permet notamment de lutter contre la contrefaçon de médicaments en assurant la traçabilité des produits et en sécurisant les données des patients. En effet, les transactions enregistrées sur la blockchain sont immuables et horodatées, ce qui garantit l'intégrité des informations et permet de suivre un produit de

manière fiable. Par ailleurs, la blockchain peut également améliorer la gestion des données médicales en offrant une plateforme sécurisée et transparente pour le partage des informations entre les différents acteurs de la santé. La blockchain offre ainsi un cadre transparent et efficace pour améliorer les services de santé tout en réduisant les coûts de traitement[12].

Les chaînes d'approvisionnement (Supply chain) :

Les applications de la technologie blockchain peuvent améliorer la transparence et la responsabilité dans les systèmes de gestion de la chaîne d'approvisionnement. Une fois les données de suivi entrées dans un grand livre blockchain, elles deviennent immuables. La blockchain renforce la confiance entre les fournisseurs de la chaîne, car tous peuvent suivre les expéditions, les livraisons et les progrès. Elle élimine les auditeurs intermédiaires, ce qui augmente l'efficacité et réduit les coûts, et les fournisseurs peuvent effectuer leurs propres vérifications à tout moment. La blockchain peut améliorer la mesure de la qualité des produits pendant leur transport. Par exemple, en analysant simplement les informations sur le parcours d'expédition et la durée d'un produit, les acteurs de la chaîne d'approvisionnement peuvent déterminer si un produit n'était pas au bon endroit ou s'il a été stocké trop longtemps. Ces problèmes sont critiques pour les produits réfrigérés, qui nécessitent une manipulation plus spéciale et plus soigneuse. De cette manière, les solutions basées sur la blockchain peuvent être utilisées pour garantir l'authenticité et la qualité des produits [13].

Les organisation autonomes décentralisées (DAO) :

Le développement rapide de la technologie blockchain a donné naissance à l'émergence de ce qu'on appelle l'Organisation Autonome Décentralisée, parfois appelée Corporation Autonome Décentralisée (DAC), qui est une nouvelle forme d'organisation dont les règles de gestion et opérationnelles sont généralement encodées sur la blockchain sous forme de contrats intelligents, et qui peut fonctionner de manière autonome sans contrôle centralisé ou intervention d'un tiers. Les DAO sont censés renverser le modèle traditionnel de gestion

hiérarchique et réduire de manière significative les coûts de communication, de gestion et de collaboration des organisations. Cependant, les DAO doivent encore relever de nombreux défis, tels que les problèmes de sécurité et de confidentialité, le statut juridique non clair, etc [14].

Nous allons étudier les DAOs plus en détails dans le prochain chapitre.

Energy Industry :

La blockchain est largement utilisée dans le secteur de l'énergie, notamment dans les microgrids. Ces derniers sont des réseaux électriques locaux qui intègrent et gèrent différentes sources et charges d'énergie pour améliorer l'efficacité et la fiabilité de la production et de la consommation d'énergie. La blockchain facilite, enregistre et valide les transactions d'achat et de vente d'énergie dans les microgrids, permettant aux utilisateurs de produire et de vendre leur excédent d'énergie au réseau.

À plus grande échelle, la blockchain est également utilisée dans les réseaux électriques intelligents pour faciliter le commerce de l'énergie. Elle permet une surveillance sécurisée de la consommation et du commerce d'énergie sans nécessiter d'intermédiaire central. Les contrats intelligents sont utilisés pour garantir la programmation des niveaux de flexibilité de puissance, la validation des accords de réponse à la demande et l'équilibre entre les besoins en énergie et la génération [12].

Stock Market :

La technologie blockchain pourrait résoudre les problèmes des systèmes de marché fragmentés, tels que l'interopérabilité, la confiance et la transparence. En raison du rôle des intermédiaires, du processus réglementaire et de la compensation des transactions opérationnelles, il faut plus de 3 jours pour finaliser toutes les transactions. Par conséquent, les participants au marché boursier, tels que les traders, les régulateurs, les courtiers et la bourse, traversent un processus fastidieux. La blockchain pourrait être la solution à cet égard. Elle peut rendre la bourse plus optimale grâce à la décentralisation et à l'automatisation. En

éliminant les intermédiaires et en accélérant les règlements des transactions, la blockchain peut contribuer à réduire les coûts. De plus, la technologie peut offrir une utilisation viable dans le règlement et la compensation des transactions tout en simplifiant la paperasserie monotone du commerce et du transfert de propriété juridique ainsi que le processus post-transaction sécurisé. En introduisant des contrats intelligents, la blockchain atténue le besoin d'un régulateur tiers en agissant en tant que régulateur pour toutes les transactions [12].

Identify Management :

Dans le monde réel, l'identité personnelle peut être vérifiée à l'aide de documents d'identité tels qu'un permis de conduire, une carte d'identité nationale et un passeport. Cependant, il n'existe pratiquement aucun système équivalent efficace pour sécuriser les identités en ligne. La blockchain pourrait offrir une approche pour contourner ce problème. Cette technologie peut être utilisée pour créer une plateforme protégeant l'identité d'un individu contre le vol ou réduisant les activités frauduleuses. La blockchain peut permettre aux individus de créer une identité chiffrée, ne nécessitant ni nom d'utilisateur ni mot de passe, tout en offrant plus de fonctionnalités de sécurité et de contrôle sur l'accès à leurs informations personnelles. En combinant la vérification d'identité avec le principe décentralisé de la blockchain [12].

1.8 Conclusion :

La technologie blockchain offre une solution innovante pour le stockage et la transmission sécurisés des informations. Dans ce chapitre nous avons examiné les principes de sécurité fondamentaux de la blockchain et nous avons également présenté les concepts clés de la blockchain, tels que les transactions, les blocs, les nœuds, ensuite nous avons discuté des mécanismes de consensus ainsi que leurs types. Enfin, nous avons discuté des applications pratiques de la blockchain, illustrant comment elle garantit la sécurité, la transparence et la fiabilité des transactions numériques.

Chapitre 2

Les organisations décentralisé autonomes

2.1 Introduction

Dans ce chapitre , nous débuterons par la présentation de la notion fondamentale d'Organisations Autonomes décentralisée, leur génesis et leur définition.
Nous aborderons également le processus d'exécution des DAO, qui implique des étapes cruciales telles que les jetons de transaction, exécution autonome ..ect. Malgré leurs avantages, les DAO font face à plusieurs défis. Nous examinerons en détail le problème de sécurité, illustré par l'incident du projet The DAO, ainsi que les préoccupations liées à la confidentialité des trésoreries des DAO et les risques de sécurité associés..
Enfin, nous aborderons les différentes techniques de confidentialité dans la blockchain, en mettant en lumière l'importance des preuves de connaissance nulle (zero-knowledge proofs) pour garantir la confidentialité des transactions et des données dans un environnement transparent et vérifiable

2.2 La notion fondamental de DAO :

En général, le terme "organisation autonome décentralisée" ("DAO") fait référence à un réseau distribué de parties prenantes qui participent à une structure organisationnelle sans se faire confiance ou même se connaître. Les parties prenantes n'ont pas de contrats d'organisation formels ou d'emploi, mais sont plutôt régies par des dispositions de contrats intelligents qui existent dans le code logiciel et une structure qui existe sur le réseau respectif.

Le réseau Bitcoin a été qualifié de "première vraie DAO". Le terme "DAO" a également été utilisé dans un livre blanc rédigé par le fondateur et directeur de la technologie de Slock.it, Christoph Jentzsch. Le livre blanc décrit la première implémentation de code DAO basé sur des contrats intelligents sur le réseau Ethereum, conçu pour "automatiser la gouvernance organisationnelle et la prise de décision". L'objectif novateur du livre blanc de Jentzsch était de partager un exemple d'une méthode qui permet aux participants de maintenir un contrôle direct en temps réel des fonds contribués et formaliser, automatiser et faire respecter les règles de gouvernance à l'aide de logiciels". Dans cette note, le terme "DAO" fait référence à une organisation distribuée dont la gouvernance est largement automatisée et ancrée dans le code informatique [15]**.**

2.2.1 Genèse des DAO :

Le concept d'une Entité Décentralisée Autonome (DAO) trouve ses origines dans un article de blog de Daniel Larimer en 2013, qui proposait que les cryptomonnaies puissent fonctionner comme des DAC, où le code source représente les statuts et les détenteurs de jetons sont les actionnaires.
Stan Larimer, père de Daniel, a développé cette idée en décrivant un DAC comme une entité gérée par un ensemble de règles commerciales incorruptibles, exécuté indépendamment de l'intervention humaine, avec des jetons accordant des droits aux bénéfices et aux votes.

Vitalik Buterin a ensuite approfondi cette notion dans des articles de blog, explorant les défis techniques et les applications potentielles des DAC. Il a finalement introduit le terme "Organisation Autonome Décentralisée" dans le Livre Blanc d'Ethereum pour décrire des contrats intelligents à long terme qui codent les statuts d'une organisation. Buterin a distingué les DAC des DAO en soulignant que les DAC étaient des entités à but lucratif, tandis que les DAO étaient à but non lucratif, bien que de l'argent puisse être gagné en participant à leur écosystème. Daniel Larimer a par la suite adopté le terme DAO, abandonnant le terme DAC, et a considéré BitShares comme le premier DAO en raison de ses caractéristiques [16].

Le premier projet : **“The DAO”** a été lancé en avril 2016 par un groupe de programmeurs. Il fonctionnait comme un fonds de capital-risque décentralisé, permettant aux contributeurs de voter sur des projets proposés en utilisant des jetons acquis lors d'une Offre Initiale de Jetons . Cependant, en juin 2016, une erreur dans le code de The DAO a permis à un attaquant de dérober une grande partie de ses fonds. Étant donné que The DAO était le plus grand projet sur Ethereum à ce moment-là, la Fondation Ethereum a décidé d'agir. Après plusieurs jours de discussions, la fondation a finalement décidé d'aller de l'avant avec un hard fork et de restituer les fonds volés aux investisseurs de The DAO. Cependant, cette décision a soulevé des préoccupations concernant l'immutabilité des enregistrements de la blockchain.

Malgré cet incident, le concept de DAO continue d'être exploré, bien qu’avec une prise de conscience accrue des risques de sécurité associés aux contrats intelligents[17].

2.2.2 Définition DAO :

Un DAO, ou Organisation Autonome Décentralisée est une entité native d'internet sans gestion centralisée qui est régulée par un ensemble de règles automatiquement exécutables sur une blockchain publique, et dont le but est de prendre vie et d'inciter les gens à accomplir une mission commune partagée.

Une autre définition indique que : "Un DAO est une organisation dont les opérations essentielles sont automatisées conformément aux règles et principes définis dans un code(contrat intelligent) sans implication humaine. Un DAO est une coordination novatrice, évolutive et auto-organisée sur la blockchain, contrôlée par des contrats intelligents" . En bref, un DAO peut être défini comme des personnes ayant des objectifs communs qui se regroupent sous une infrastructure blockchain qui impose un ensemble de règles partagées.

En général, les membres d'un DAO sont enregistrés, chacun avec une adresse unique. Ils possèdent également une quantité de jetons de 'gouvernance' liés à cette adresse, qui sont généralement requis pour la participation, et peuvent jouer un rôle dans le système de prise de décision du DAO. Il est également courant que les DAO gèrent des ressources, par exemple des cryptomonnaies.

Les membres du DAO peuvent décider comment les allouer grâce à un système de décision, L'activité du DAO est enregistrée dans la blockchain et, par conséquent, implique un coût. Valider et confirmer les transactions sur la blockchain Ethereum nécessite un certain travail, appelé gaz (payé en cryptomonnaie). Ce travail est effectué par les mineurs de la blockchain afin d'inclure les transactions dans un bloc. Le gaz se traduit finalement en argent et la quantité de gaz dépend de la taille et du type de chaque transaction.

Par conséquent, il est attendu que l'activité du DAO soit conditionnée par cela, car les utilisateurs doivent payer de petites quantités de cryptomonnaie s'ils veulent que leur opération soit exécutée[17].

2.3 Processus d'exécution des DAO :

Les décisions sont prises et mises en œuvre au sein de l'organisation autonome décentralisée comme suivants :

- **Jetons de transaction :** Les DAO utilisent des jetons comme représentation de la propriété ou de la valeur dans le réseau. Ces jetons peuvent être utilisés pour récompenser les membres du DAO pour leur contribution et peuvent également être utilisés pour voter sur les décisions du DAO.

- **Exécution autonome :** Un DAO est conçu pour fonctionner de manière autonome, sans intervention humaine directe une fois qu'il a été lancé. Les décisions sont prises en fonction de règles prédéfinies dans le code du DAO, ce qui garantit un fonctionnement transparent et impartial.

- **Consensus** :Les décisions au sein d'un DAO sont prises par consensus. Cela signifie que toutes les parties prenantes doivent être d'accord pour qu'une décision soit mise en œuvre. Cela garantit que les décisions sont prises de manière démocratique et transparente.

- **Contractants :** Pour accomplir les tâches et atteindre les objectifs du DAO, des contractants externes peuvent être engagés. Ces contractants peuvent être des fabricants, des développeurs, ou d'autres entrepreneurs, et sont sélectionnés en fonction des décisions prises par les membres du DAO.

- **Propositions :** Tout membre du DAO peut proposer une action ou une décision à prendre. Les propositions peuvent inclure des changements dans les règles du DAO, des investissements à réaliser, ou d'autres actions. Les propositions peuvent nécessiter un dépôt monétaire pour éviter un nombre excessif de propositions et garantir que seules les propositions sérieuses sont soumises au vote.

- **Vote :** Une fois qu'une proposition a été faite, les membres du DAO votent pour décider de son approbation ou de son rejet. Une majorité simple est généralement nécessaire pour qu'une proposition soit acceptée [18].

2.4 Fonctionnement des DAO :

Une DAO peut agir comme une plateforme, où les membres interagissent selon un protocole open source auto-exécutoire. Ceux qui souhaitent promouvoir l'objectif de la DAO (ou du moins optimiser la valeur de la DAO) achètent les jetons de la DAO. Ces jetons donnent des droits de vote aux membres, qui peuvent alors voter sur les propositions soumises par d'autres membres de la DAO (par exemple, entreprendre un nouveau projet). Il est dans l'intérêt de tous les détenteurs de jetons de la DAO que seules les propositions bénéfiques qui

optimiseront la valeur de la DAO soient approuvées. Si la proposition est approuvée, elle sera enregistrée dans la blockchain. La rémunération pour l'exécution de la proposition sera généralement codifiée dans un contrat intelligent, de sorte que la compensation ne sera accordée que lorsque les proposants auront tenu leur promesse.

Les DAO avec cette architecture ont le potentiel de conserver un fort élément humain, car les membres de la DAO (qui peuvent être des humains ou des machines) votent toujours sur les décisions et proposent leurs propres propositions. Malgré cet élément humain, avec les règles de gouvernance codifiées dans des contrats intelligents, les décisions de gouvernance pertinentes sont toujours exécutées automatiquement sans intervention manuelle, contournant ainsi le besoin d'une entité décisionnelle centrale.

À l'extrémité opposée du spectre autonome se trouvent les DAO qui intègrent l'intelligence artificielle pour fonctionner entièrement de manière autonome sur une blockchain. Les activités de ce type de DAO sont entièrement déterminées par le protocole de la blockchain et le code des contrats intelligents de la DAO. Comme une DAO peut utiliser des jetons numériques pour déclencher des contrats intelligents indépendamment, finalement, avec un nombre suffisant de jetons numériques pour payer un réseau blockchain pour les ressources dont elle a besoin, une DAO peut fonctionner indéfiniment sans contrôle humain. Les humains peuvent encore contribuer des fonds à la DAO en échange de jetons numériques (et d'une part des bénéfices de la DAO), ou interagir avec les DAO en payant pour son service. Ces DAO peuvent être construits de deux manières distinctes[14].

2.5 Les différents Défis et problème de DAO :

2.5.1 Problème de sécurité :

En raison de la nature infalsifiable de la blockchain, il est difficile de modifier un DAO ou les contrats intelligents qui le sous-tendent une fois qu'il est déployé sur la blockchain. Ainsi, les attaquants peuvent profiter d'une faille dans le contrat intelligent pour réaliser un profit.

Pour comprendre ce genre de problème, nous allons expliquer le problème survenu pour le DAO “The DAO”.

2.5.1.1 The DAO :

Le DAO a été lancé le 30 avril 2016 (par plusieurs soumissions "anonymes" associées à DAOhub, qui ont exécuté le bytecode open-source sur la blockchain Ethereum), avec une période de financement ou de "création" de 28 jours . À la fin de la période de financement (le 28 mai 2016), le DAO est devenu opérationnel avec l'équivalent d'environ 250 millions de dollars de financement, battant tous les records de financement participatif. Environ 10 000 à 20 000 personnes (estimation) ont investi dans le DAO, contribuant à hauteur de 11 994 260,98 jetons Ethereum (appelés ether, ou ETH), ce qui représentait environ 14 % de l'offre totale d'ETH.

Cependant, peu de temps après la période de "débat" minimale de deux semaines, le 17 juin 2016, le code du DAO a été exploité par un individu inconnu. Cette exploitation a utilisé un comportement non intentionnel de la logique du code pour vider rapidement le fonds de millions de dollars en jetons ETH il a retiré des fonds du DAO vers un enfantDAO, ce qui a entraîné le vol de 3,5 millions d'ETH (environ 50 millions de dollars).

2.5.1.2 Confidentialité de trésorerie :

Un DAO est une forme de partenariat où un groupe de personnes se réunit pour atteindre un objectif commun . Récemment, un DAO appelé ConstitutionDAO s'est formé dans le but d'acheter l'une des dernières copies physiques de la Constitution américaine. Le DAO a levé un montant étonnant de 40 millions de dollars en moins d'une semaine pour placer une offre. Cependant, le DAO a rapidement été surenchéri par un investisseur individuel qui a placé une offre à 43,17 millions de dollars, juste un peu au-dessus des actifs de ConsutitutionDAO, assurant ainsi que l'investisseur remportait l'enchère.

Le problème avec l'état actuel des DAO réside dans le fait que leurs fonds sont publics. Lorsque ConstitutionDAO levait des fonds, l'ensemble de son bilan était visible par le public. Cette propriété peut être préjudiciable aux DAO et entraver leur capacité à participer à certains types d'enchères. En particulier, les DAO ne peuvent pas participer aux enchères à soumission cachée précisément parce que leurs trésors sont publics. Dans le cas de ConstitutionDAO, le fait d'avoir un trésor public a permis à d'autres soumissionnaires de connaître son offre maximale et de surenchérir [19]**.**

2.5.1.3 Risques de sécurité :

Comme l'a clairement montré “The DAO”, les DAO et les technologies blockchain en général restent susceptibles aux menaces de sécurité qui pourraient finalement avoir des conséquences graves si elles étaient exploitées. Les contrats intelligents, un aspect central des DAO, peuvent être considérés comme une vulnérabilité car ils peuvent être difficiles à modifier une fois qu'ils sont déployés sur la blockchain . Les attaquants peuvent profiter des failles dans les contrats, comme cela a été le cas pour “The DAO”, et il ne serait pas facile de les inverser en raison de la nature résistante à la manipulation de la blockchain .

De plus, les auteurs de l’article [20] ont conclu que la technologie blockchain reste confrontée à des risques de sécurité, ce qui rend les DAO susceptibles aux vulnérabilités. Ces risques incluent la criminalité, l'inexactitude et la sous-optimisation dans les contrats . De plus, la technologie blockchain dans son ensemble pourrait également être exposée à différents types d'attaques . Comme les DAO sont construites autour de la technologie blockchain, le risque d'attaques sur la blockchain pourrait potentiellement rendre les DAO vulnérables[21].

2.5.2 Problèmes de gouvernance :

Bien que les DAO aient l'intention de fonctionner comme des organisations autonomes et auto-gouvernées, la réalité est qu'à ce stade, elles restent susceptibles d'être affectées par des forces externes et, dans certains cas, même dépendantes d'acteurs externes . Dans le cas de 'The DAO', le manque d'autorité centrale au sein de l'organisation a probablement rendu difficile la réaction et la tentative de résolution rapide de la situation critique.

Ce manque d'autorité a conduit à la nécessité de faire intervenir des parties externes pour organiser la gouvernance. Les auteurs de [22] explique qu' au moment où les acteurs externes sont intervenus pour gérer les conséquences de la vidange des fonds de 'The DAO', "la vision des structures de gouvernance futures" aspirées par les DAO s'est effondrée, car l'intervention externe reflète la pensée des organisations traditionnelles. En d'autres termes, bien que les activités menées par les DAO soient gérées selon des principes d'auto-gouvernance, elles restent dépendantes de forces externes et peuvent finalement être influencées par elles .

L'auteur [22] indique également que la réalisation d'un 'hard fork' pour résoudre le problème a été perçue comme une forme de gouvernance centralisée où un organe décisionnel autoritaire a décidé de résoudre un problème. De ce point de vue, la décision d'un 'hard fork' semble être en conflit direct avec l'idée de prise de décision décentralisée préconisée par les DAO, suggérant qu'à l'heure actuelle, de telles organisations ne sont pas vraiment libres de la prise de décision centralisée.

2.5.2.1 Problème de vote :

Un autre problème de gouvernance concerne les mécanismes de vote des DAO . Bien qu'une DAO agisse de manière autonome en étant gouvernée uniquement par les membres de sa communauté, elle peut ne pas nécessairement être un système équitable, car le pouvoir de vote au sein de l'organisation peut ne pas être égal.

Alors que les DAO tentent de mettre en œuvre un système démocratique avec des humains collaborant et agissant en tant que décideurs grâce à des systèmes de vote, les utilisateurs sont

souvent anonymes et travaillent sous des pseudonymes, ce qui signifie qu'une seule personne peut créer plusieurs alias et donc voter plusieurs fois Dans ce scénario, une seule personne émettant plusieurs bulletins de vote entraîne un système qui n'est pas vraiment démocratique, car une personne peut potentiellement avoir une influence plus importante sur la DAO que les autres membres de la communauté.

Cependant, cette situation est atténuée dans les DAO qui fonctionnent selon des principes de travail pour gagner tels que Colony où les membres gagnent en influence en fonction de leurs contributions. Néanmoins, comme toutes les DAO n'opèrent pas de cette manière, le problème reste une préoccupation [21].

2.5.3 Opportunités :

Il est encore trop tôt pour prédire comment les DAO évolueront et si elles atteindront finalement leur plein potentiel. L'auteur de [23], par exemple, souligne que la communauté de la recherche est confrontée à des questions concernant la pertinence des DAO pour les formes organisationnelles existantes, les structures de gouvernance basées sur la blockchain, l'élaboration des politiques et les nouveaux cas d'utilisation également pour les secteurs industriels traditionnels . En examinant certaines des utilisations actuelles des DAO et en examinant ses principes de décentralisation.

On peut faire quelques suggestions sur les industries qui pourraient en bénéficier. Récemment, bien qu'elles ne soient pas devenues entièrement décentralisées, certaines entreprises ont adopté des modèles où les employés se voient accorder plus de pouvoir de décision et de liberté pour prendre en charge et gérer les tâches comme ils l'entendent . Permettre aux employés cette liberté d'auto-organisation sans avoir besoin d'impliquer la haute direction suggère que certaines organisations sont ouvertes à une approche plus décentralisée qui pourrait ouvrir la voie à des opportunités futures pour les DAO.

De plus, certaines des fonctionnalités offertes par les DAO pourraient bénéficier à des industries spécifiques. Alors que les DAO précédents étaient pertinents dans l'espace crypto ,

un changement a récemment été observé vers des secteurs plus traditionnels . Dans ce qui suit, les avantages potentiels de ces DAO par rapport aux organisations traditionnelles sont examinés. Ensuite, les industries où les DAO pourraient avoir des motifs d'application sont explorées[21].

2.6 Technique de confidentialité utilisées dans la blockchain :

2.6.1 Mixing(Mixeur) :

Les services de mélange (ou tumblers) ont été conçus pour empêcher que les adresses des utilisateurs soient liées. Le mélange, littéralement, consiste en un échange aléatoire des pièces de l'utilisateur avec celles d'autres utilisateurs. En conséquence, pour l'observateur, la propriété de leurs pièces est obscurcie. Cependant, ces services de mélange ne protègent pas contre le vol de pièces. Dans cette section, nous décrivons deux de ces services de mélange et analysons leurs propriétés en matière de sécurité et de confidentialité.

2.6.2 Mixcoin :

Mixcoin a été proposé par [24] en 2014. Il offre un paiement anonyme en Bitcoin et en crypto-monnaies similaires au bitcoin. Pour se défendre contre les adversaires passifs, Mixcoin étend l'ensemble d'anonymat pour permettre à tous les utilisateurs de mélanger des pièces simultanément. Pour se défendre contre les adversaires actifs, Mixcoin fournit un anonymat similaire aux mélanges de communication traditionnels. De plus, Mixcoin utilise un mécanisme de responsabilité pour détecter le vol, et il montre que les utilisateurs utilisent Mixcoin rationnellement sans voler de bitcoins en alignant les incitations.[25].

2.6.3 Zero-knowledge proofs systems :

Une preuve à divulgation nulle (zero-knowledge proof en anglais) est un protocole cryptographique entre un prouveur et un vérificateur, où le prouveur démontre la vérité d'une assertion sans révéler aucune information supplémentaire au vérificateur, autre que la validité de l'assertion. Pour être considérée comme une preuve à divulgation nulle, la preuve doit satisfaire trois propriétés principales

1. Complétude : Le vérificateur doit être convaincu de la vérité de l'assertion si le prouveur est honnête et suit le protocole correctement. En d'autres termes, si l'assertion est vraie, le vérificateur doit accepter la preuve.

2. Correction : Si l'assertion est fausse, alors aucun prouveur malveillant ne devrait pouvoir convaincre le vérificateur que l'assertion est vraie, sauf avec une probabilité négligeable.

3. Secret partagé : Après avoir terminé le protocole, le vérificateur n'apprend rien d'autre que la vérité de l'assertion. En d'autres termes, le protocole ne révèle aucune information sur les secrets utilisés dans la preuve [26].

Prenons deux exemples pour illustrer le concept :

Exemple 1:

Imaginez qu'il y ait un prouveur et un vérificateur. Le prouveur veut convaincre le vérificateur qu'il connaît quelque chose sans lui en dire plus. Cela peut sembler étrange au premier abord, mais voici une analogie :
Imaginez qu'il y ait une photo avec un macareux caché parmi plusieurs pingouins. Le prouveur sait où se trouve le macareux, mais il ne veut pas donner cette information au vérificateur. Il lui montre simplement une preuve en cachant la photo derrière une affiche. Le

vérificateur regarde par un trou et voit le macareux. Il ne sait pas où se trouvait la photo, mais il est convaincu que le prouveur connaissait bien l'emplacement du macareux.

2.7 Conclusion :

En conclusion, ce chapitre a posé les bases pour une compréhension approfondie des Organisations Autonomes Décentralisées (DAO). Nous avons exploré leur genèse, leur définition, ainsi que leur fonctionnement .Nous avons également abordé le processus d'exécution des DAO, en mettant en lumière des étapes cruciales telles que la gestion des jetons de transaction et l'exécution autonome des décisions prises. Malgré leurs avantages, les DAO font face à plusieurs défis, notamment en matière de sécurité et de confidentialité.. Enfin, nous avons exploré différentes techniques de confidentialité dans la blockchain, en présentant plusieurs techniques comme les mixer et les preuves de connaissance nulle (zero-knowledge proofs) pour assurer la confidentialité des données.

Chapitre 3

Notre solution pour une trésorerie confidentielle dans un DAO

3.1 Introduction

Dans ce chapitre nous allons discuter sur le problème de la confidentialité de la trésorerie du DAO en expliquant par un exemple du Constitution DAO le plus célèbre qui met en lumière les défis liés à la transparence financière des DAO

Le principal défi des DAO réside dans la visibilité des ressources financières qui sont gérées dans une blockchain publique Cela signifie que tout le monde peut connaître le montant total de la trésorerie du DAO. De plus, lors de la contribution des participants en envoyant les fonds au DAO, toutes les informations de cette contribution sont enregistrées dans les transactions de la blockchain et sont publiquement accessibles. Cette transparence peut poser des risques pour les DAO dans certaines situations, notamment lorsqu'ils participent à des enchères, car ils pourraient perdre l'enchère, compromettant ainsi la sécurité, la confidentialité et la gestion des fonds au sein du DAO. Ces risques comprennent :

Divulgation d'informations sensibles :
Les détails des transactions et les soldes de chaque contribution seront publiquement accessibles, révélant ainsi des informations sensibles sur les activités financières du DAO, telles que les montants des transactions, les adresses de portefeuille et les bénéficiaires des fonds.

Risques de sécurité :

La transparence totale des activités du DAO peut attirer des attaquants malveillants cherchant à exploiter des failles dans les transactions ou à cibler les fonds du DAO.

Manque de confidentialité des participants :

Les contributeurs au DAO peuvent craindre que leurs informations financières ne soient pas privées, ce qui peut les dissuader de participer ou les inciter à se retirer.

3.2 Une étude de cas avec Constitution DAO :

Un DAO est une forme de collaboration où un groupe de personnes se réunit pour atteindre un objectif commun. Récemment, un DAO nommé Constitution DAO s'est formé avec l'objectif d'acquérir une des dernières copies physiques de la Constitution américaine. En moins d'une semaine, ce DAO a réussi à lever la somme impressionnante de 40 millions de dollars pour faire une offre. Cependant, il a été rapidement surenchéri par un investisseur individuel proposant 43,17 millions de dollars, juste au-dessus des fonds collectés par Constitution DAO, assurant ainsi la victoire de l'investisseur à l'enchère.

Le problème avec les DAO, dans leur forme actuelle, réside dans le caractère public de leur trésorerie. Lorsque Constitution DAO collectait des fonds, l'intégralité de son bilan était visible au public. Cette transparence peut être préjudiciable aux DAO, limitant leur capacité à participer à certains types d'enchères, notamment les enchères scellées. Dans le cas de Constitution DAO, la visibilité de sa trésorerie a permis aux autres enchérisseurs de connaître son enchère maximale et de la surpasser.

L'expérience de Constitution DAO met en évidence les conséquences négatives de la divulgation des ressources financières d'un DAO.

3.3 Problématique traitée dans notre Document :

Après avoir exposé le problème survenu lors de l'enchère à laquelle Constitution DAO a participé, nous allons proposer une solution à la problématique suivante : Comment garantir que la trésorerie d'une organisation décentralisée autonome (DAO) peut être masquée, tout en lui permettant de participer à une enchère et de prouver qu'elle possède une somme d'argent supérieure à la valeur de son enchère, sans dévoiler le montant total de ses fonds.

La solution que nous proposons repose sur le protocole Tornado Cash, développé sur la blockchain Ethereum. Ce protocole est décrit dans la section suivante afin que le lecteur puisse comprendre le fonctionnement du protocole que nous proposons.

3.4 Protocole de Tornado Cash :

Tornado cash est un mixeur basé sur un protocole de Zero Knowledge Proof construit sur la blockchain Ethereum.. Il a été lancé en mai 2020 et a été assez utilisé depuis. Il permet aux utilisateurs de rendre leurs transactions anonymes en utilisant des contrats intelligents .

Tornado Cash est essentiellement une adresse sur la blockchain Ethereum. Imaginez qu'Alice a son propre compte sur la blockchain Ethereum. À tout moment, elle peut envoyer des fonds aux contrats de Tornado Cash. Dans l'exemple ci-dessous (figure 8), elle envoie 300 ETH (l'unité de crypto-monnaie d'Ethereum) au contrat en trois transactions séparées, chacune valant 100 ETH.

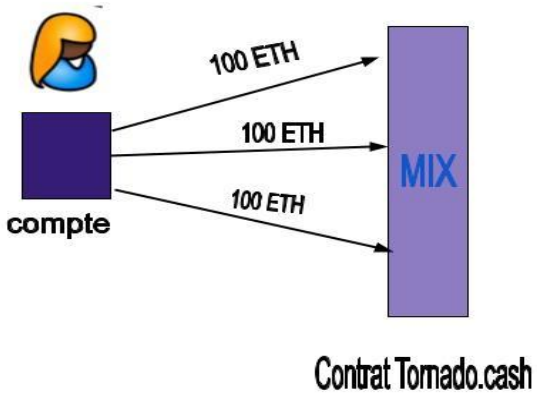

Figure 8 : Processus de Mixeur de Tornado Cash

Elle peut garder ses fonds dans le contrat Tornado Cash aussi longtemps qu'elle le souhaite. Entre-temps, d'autres personnes peuvent également envoyer leurs propres fonds au contrat. Disons que quelque temps plus tard, peut-être une semaine, un mois ou un an plus tard, Alice décide de retirer ses fonds du contrat Tornado Cash. Elle prouvera au contrat qu'elle possède 300 ETH dans le contrat et fournira une adresse où déposer ces fonds. Le contrat Tornado Cash enverra 300 ETH à l'adresse qu'elle a fournie, mais le point important est que maintenant, personne ne sait que cette adresse est affiliée à Alice. Elle a donc 300 ETH dans cette adresse qu'elle contrôle, mais personne ne sait à qui appartient cette adresse.Voir la figure 9.

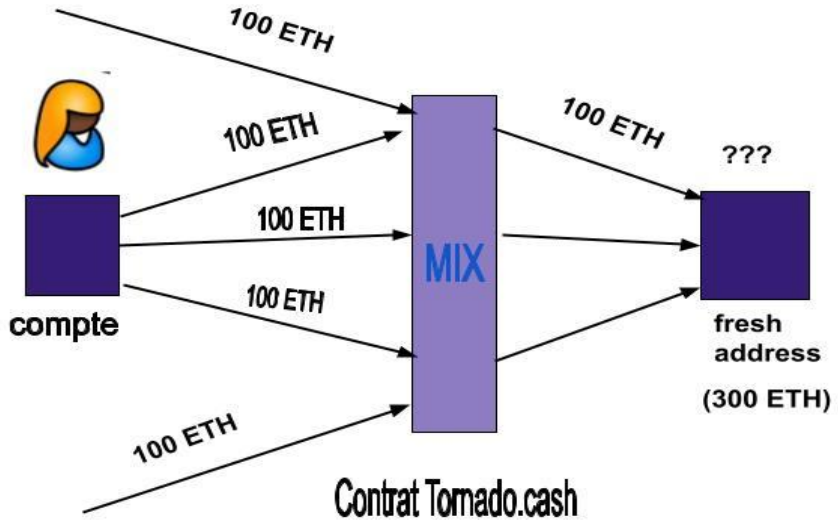

Figure 9 : le processus d'envoi d'ETH de manière anonyme

Alice peut ensuite utiliser ces fonds, par exemple, pour interagir avec un marché NFT et acheter un NFT de manière privée. Elle possédera le NFT, mais personne ne saura qui possède réellement ce NFT.

Les transactions dans Tornado cash ont le même montant, soit 100 ETH, et la raison pour cela est la confidentialité. Comme présenté dans la figure 10, Si Alice était autorisée à envoyer 105,3 ETH au contrat et à retirer plus tard 105,3 ETH du contrat, ces valeurs de transaction créeraient un lien entre la nouvelle adresse et l'adresse originale d'Alice. Pour s'assurer qu'il n'existe aucun lien entre l'adresse originale et la nouvelle adresse, nous veillons à ce que tous les montants des transactions entrant et sortant du contrat soient les mêmes, dans ce cas 100 ETH.

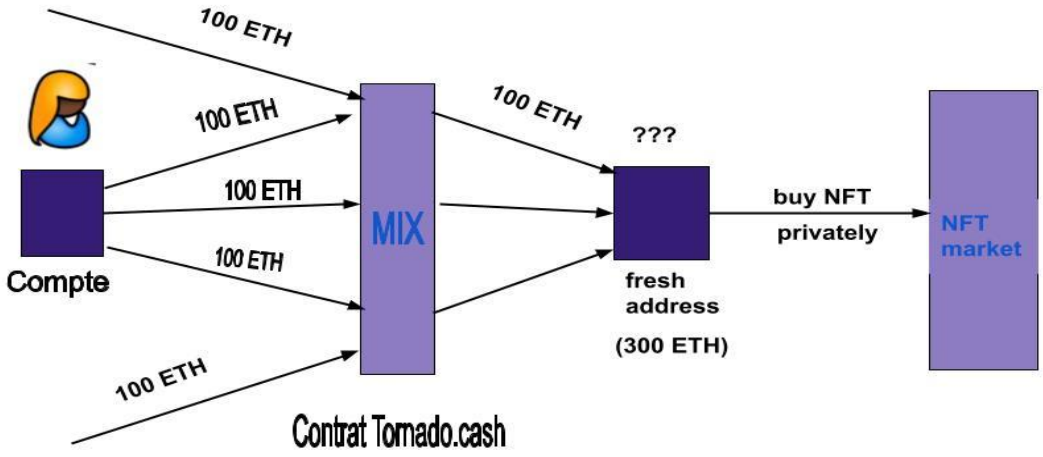

Figure 10 : Achat Privé de NFT avec Envoi Anonyme via Tornado Cash

3.4.1 Fonctionnement du contrat Tornado Cash:

Tornado Cash dispose d'un pool (contrat) pour différentes dénominations (valeurs de dépôt). Nous examinerons le pool de 100 ETH. Tous les dépôts et retraits dans ce pool seront donc de 100 ETH.

Le contrat Tornado Cash, maintient bien sûr son propre solde. Disons qu'il a une trésorerie, et disons que la trésorerie contient actuellement 300 ETH, ce qui signifie que trois pièces ont été déposées dans le contrat, chaque pièce valant 100 ETH.

En outre, le contrat Tornado Cash gère une root de Merkle, que nous appelons les racines de Merkle des pièces. Cet arbre de Merkle contient essentiellement la liste des pièces dans le contrat jusqu'à présent. Ces pièces sont insérées de gauche à droite. La première pièce va dans la feuille la plus à gauche de l'arbre, puis nous continuons d'ajouter de gauche à droite au fur et à mesure que des pièces sont ajoutées. Cet arbre a une hauteur de 20, ce qui signifie qu'il y a 2^20 feuilles au total dans l'arbre.

Si l'arbre est rempli un nouveau contrat sera créé . Au fur et à mesure que de nouvelles pièces sont ajoutées, elles sont ajoutées séquentiellement aux feuilles de l'arbre. Nous calculons alors le hachage de Merkle de l'état actuel de l'arbre, ce qui nous donne la racine de Merkle des pièces, stockée dans l'état du contrat Tornado Cash.

La preuve de Merkle est de 32 octets, donc elle prend très peu de place. Le contrat Tornado Cash maintient un "état" qui est constitué d'une variable appelée "next" qui indique où la prochaine pièce sera insérée dans l'arbre, Dans la figure 11 ci-dessous, next fait référence à la position quatre. "l'état du contrat" contient également la preuve de Merkle pour cette position. Une preuve de Merkle contient la liste des hachages de cette feuille à la racine. Le contrat stocke également ces preuves de Merkle, contenant 20 hachages, dans l'état du contrat. En outre, le contrat maintient une liste de nullifieurs nous expliquerons leur utilisation plus bas. Pour résumer, il y a dans la figure 17 ci-dessou qui représente l'état d'un contrat Tornadi cash trois pièces dans le pool, le contrat a 300 ETH et il y a deux "nullifiers" dans le contrat.

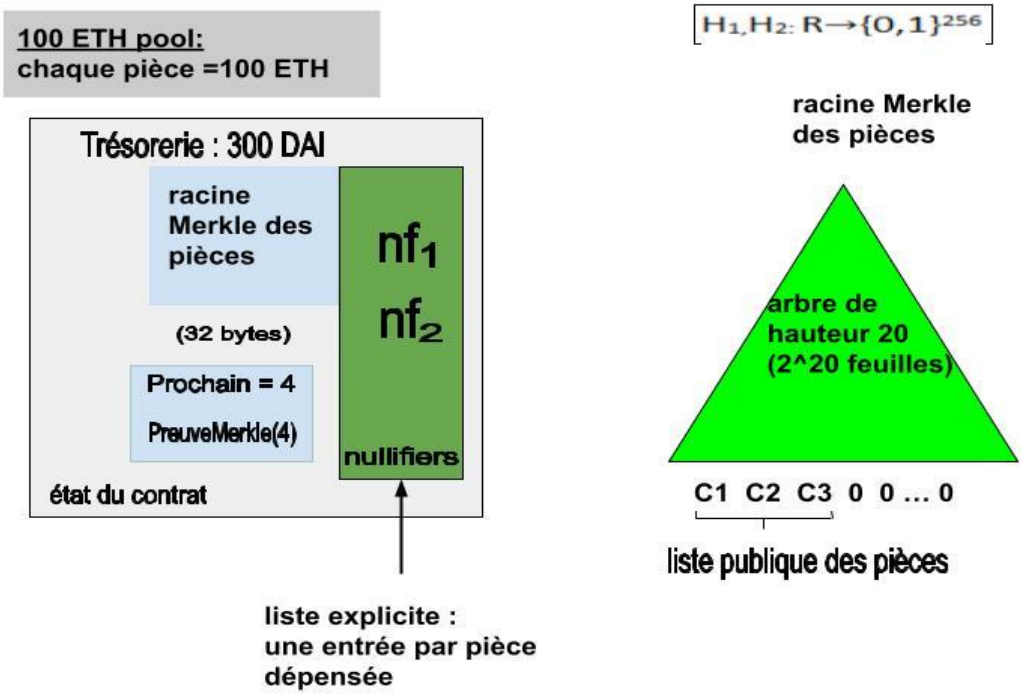

Figure 11 Structure de l'état de contrat de Tornado Cash

Pour ce qui va suivre, nous allons avoir besoin de deux fonctions de hachage que nous désignerons par h1 et h2. Elles prennent une valeur dans R et produisent une valeur de 32 octets.

Processus de dépôt:

Supposons qu'Alice veuille déposer 100 ETH dans le contrat Tornado Cash. Tout d'abord, elle choisira une clé aléatoire k et un nonce aléatoire r à partir de l'espace de random R. Ensuite, elle calculera la nouvelle pièce c4 en prenant le hachage de k et r, soit h1(k | r), ce qui donne une valeur de 32 octets. Elle inscrit cette valeur c4 dans la quatrième feuille de

l'arbre, à l'emplacement libre suivant. Puis, elle calcule la preuve de Merkle pour l'emplacement suivant, la position cinq dans l'arbre, ce qui implique la liste des hachages de cet emplacement à la racine. Cette preuve π contient 20 hachages que Alice calcule, elle enverra ses 100 ETH au contrat, en plus de la nouvelle pièce c4 et de la preuve de Merkle pour la position 5 (next +1) comme montré dans la figure 12.

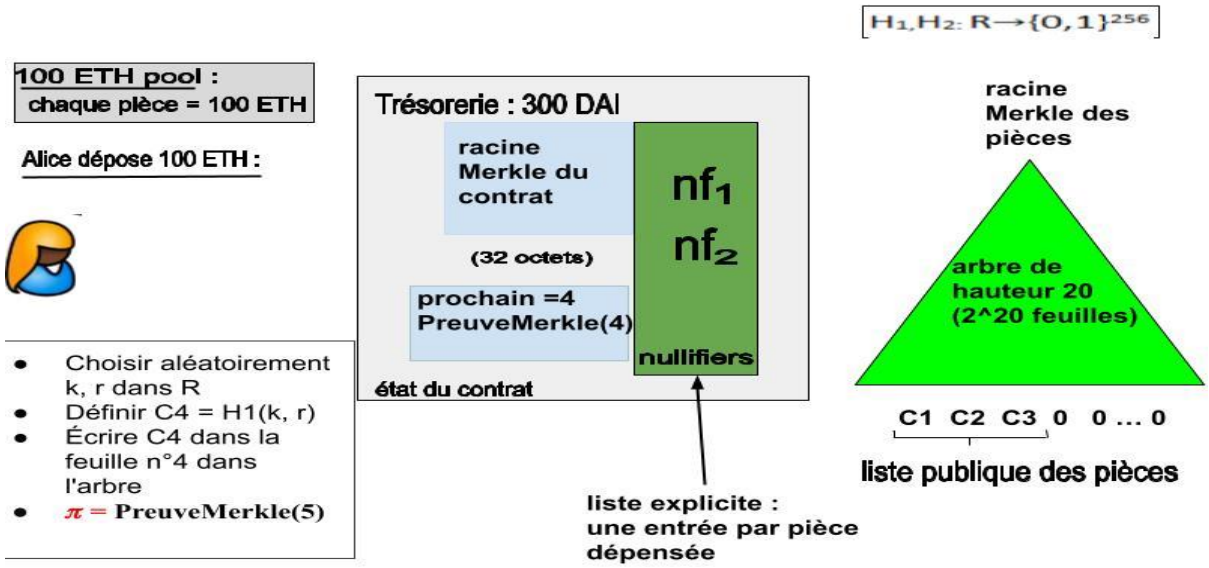

Figure 12 : Processus du Dépôt

État du contrat:

Le contrat utilise c4 et la preuve de Merkle stockée dans l'état du contrat pour calculer la nouvelle racine de Merkle, puis vérifie que la preuve de Merkle fournie par Alice pour la position 5 est correcte. Si tout est valide, le contrat met à jour son état :

- La trésorerie passe à 400 ETH (car Alice a déposé 100 ETH supplémentaires).
- La racine de Merkle est mise à jour avec c4.
- La valeur "next" est mise à jour à 5.
- La preuve de Merkle pour la position 5 est stockée dans le contrat.

Alice doit garder secrètement les valeurs k et r, appelées "note" ou ticket , correspondant aux 100 ETH qu'elle a déposés.Voir la figure 13. Elle devra par la suite utiliser ce ticket afin de récupérer ses fonds.

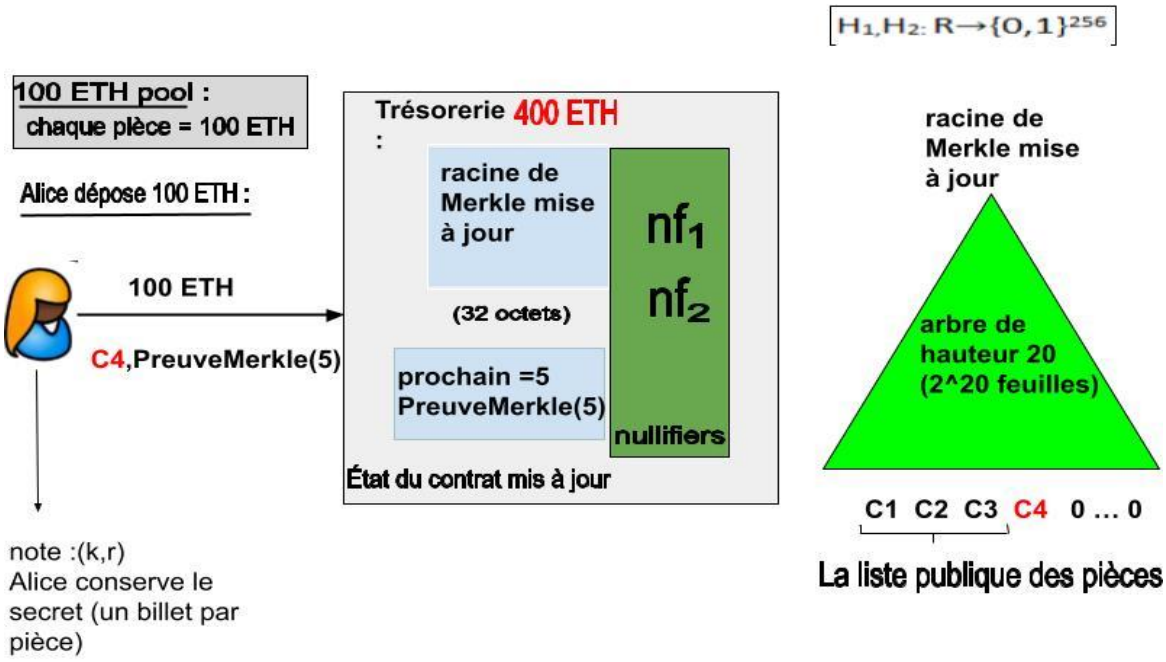

Figure 13: La mise à jour du contrat après le dépôt

Pour résumer, à chaque dépôt dans le contrat, nous créons une nouvelle pièce sur l'arbre (une nouveau coin dans l'une des feuilles vides de l'arbre), et ces pièces sont ajoutées séquentiellement à l'arbre. Notons enfin, qu'un observateur qui regarde l'état du contrat peut facilement dire à quelle pièce appartient qui. L'observateur a vu qu'Alice a ajouté c4 au contrat, donc il sait que c4 appartient effectivement à Alice.

Processus de retrait:

Supposons que Bob possède la pièce numéro trois dans l'arbre. Il a une note (ticket) pour cette pièce, k' et r', et veut retirer ses 100 ETH correspondants. Il commence par calculer un nullifieur h2(k'). Ensuite, il prouve au contrat qu'il possède une note (k', r') pour une feuille dans l'arbre de Merkle, sans révéler laquelle. Il envoie ce nullifieur au contrat, qui l'ajoute à la liste des nullifieurs. Bob utilise une zero knowledge proof pour prouver qu'il possède une note valide sans révéler quelle pièce correspond à sa note. Voir La figure 14.

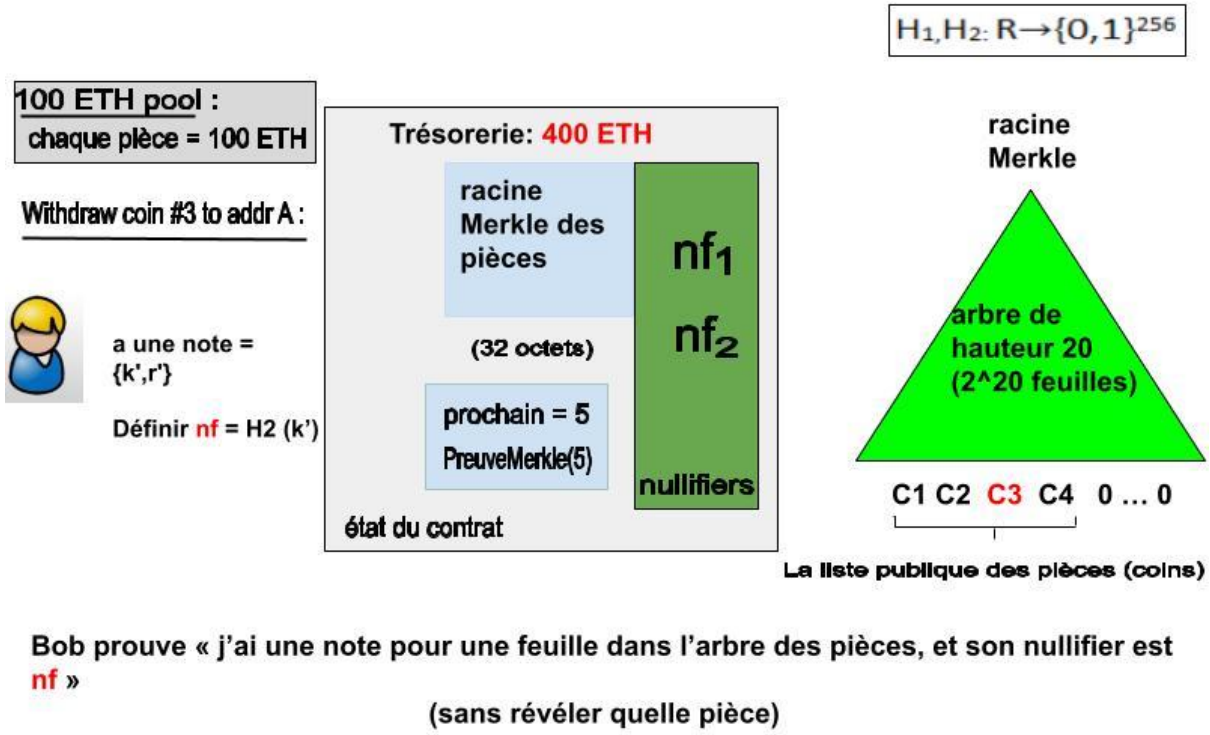

Figure 14: Processus de retirer les fonds

Le contrat vérifie la validité de la preuve et que le nullifieur n'est pas déjà dans la liste, prouvant ainsi que la pièce n'a pas été dépensée avant. Ensuite, le contrat envoie les 100 ETH à l'adresse fournie par Bob, et la trésorerie est réduite de 100 ETH .Voir la figure 15 ,Sachant que :

- Personne ne pourra retirer C3 plus tard à cause de nullifieur
- Personne ne saura que C3 a été retiré car l'adresse où les fonds sont retirés peut appartenir à n'importe ayant déjà deposé les fonds dans le contrat.

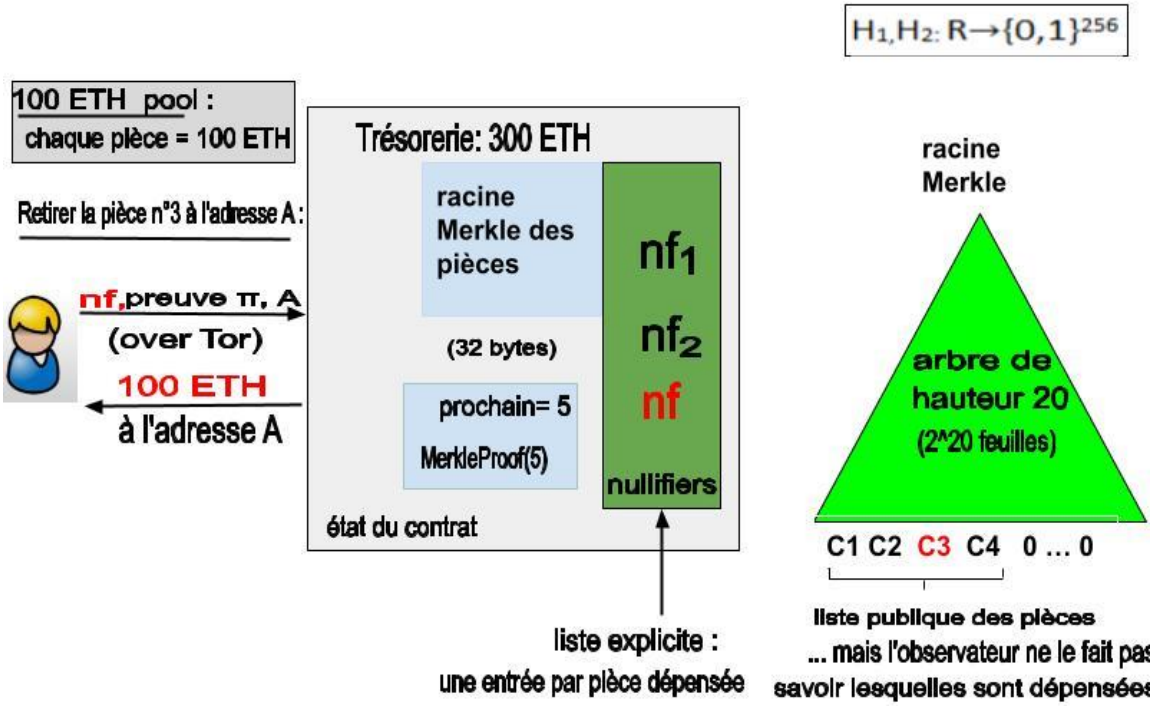

Figure 15: La mise à jour du contrat après le retrait

3.5 Notre contribution

Nous allons dans la suite de ce document présenter la solution que nous proposons afin de garantir la confidentialité de la trésorerie dans un DAO. Notre solution est basée sur une solution déjà existante. proposée par Griffin Dunaif et Dan Boneh [19]. La solution proposée par Griffin Dunaif et Dan Boneh fonctionne correctement en termes de confidentialité mais ne donne pas la possibilité à un contributeur de récupérer ses fonds s'il en a besoin. Dans la solution que nous allons proposer, Alice (le contributeur) va pouvoir fournir des fonds au DAO de son choix. Mais dans le cas où elle désire reprendre ses fonds. elle pourra le faire. A condition bien sûr, que les fonds ne soient pas déjà dépensés par le gestionnaire du DAO.

3.5.1 La solution proposée par Griffin Dunaif et Dan Boneh :

Le plan consiste à déployer un seul contrat principal sur la chaîne Ethereum pour gérer plusieurs DAOs, ce qui facilite la création et la gestion de ces DAOs. Ce contrat principal agit comme une plateforme centrale offrant les fonctionnalités nécessaires pour créer et interagir avec les DAOs.

Chaque DAO créé sur cette plateforme aura un gestionnaire DAO désigné ou un groupe désigné de gestionnaires. Ces gestionnaires sont responsables de la gestion et de la gouvernance du DAO, prenant des décisions telles que l'allocation des fonds et l'approbation des propositions.

La plateforme permet à quiconque d'envoyer des fonds à un DAO. Cela signifie que les utilisateurs peuvent contribuer au DAO en envoyant des tokens ou de l'Ether à l'adresse du contrat principal. Ces fonds seront ensuite attribués au DAO correspondant et pourront être utilisés selon les décisions prises par les gestionnaires.

Le plan vise également à garantir la confidentialité des opérations de dépôt et de retrait, où les opérations de dépôt et de retrait sont gérées publiquement, mais le solde restant de la DAO reste caché, contribuant ainsi à protéger la confidentialité des utilisateurs.

3.5.2 Notre proposition de solution

Dans la solution que nous avons proposée et qui est basée sur celle de Griffin Dunaif et Dan Boneh qui elle même est basée sur le protocole Tornado cash, nous utilisons deux Arbres de Merkle. Dans ce qui suit, nous allons faire référence au premier arbre de Merkle par **“Arbre gestionnaire”** et pour le second arbre, nous allons lui faire référence par **“Arbre contributeur”.**

Le protocole proposé consiste en trois étapes principales :

1 Création du DAO :

En déployant un contrat principal sur la chaîne Ethereum, le gestionnaire DAO entame la création d'un nouveau DAO. Cette étape se déroule en dehors de la chaîne Ethereum, ce qui signifie qu'elle n'implique pas de transactions directes sur la blockchain. Le gestionnaire définit les paramètres du DAO, tels que son nom et ses règles de gouvernance, via le contrat principal. Une fois que le DAO est créé, le gestionnaire publie la clé publique du contrat principal. Cela permet aux participants de contribuer au DAO en envoyant des fonds à cette adresse.

2.Dépôt :

En publiant l'adresse du contrat principale, tout participant peut envoyer les fonds à cette adresse du contrat principale mais Lors de cette opération, il n'est pas possible pour un observateur de déterminer à quel DAO spécifique les fonds sont destinés. L'adresse du DAO bénéficiaire reste confidentielle. Seul le gestionnaire DAO pourra accéder à ces fonds ultérieurement lors de l'étape de retrait.

3.Retirer les fonds :

Seul le gestionnaire du DAO à le droit de retirer les fonds. Lors de la participation dans l'enchère il donne un preuve à une tierce partie pour montrer que le solde du DAO est supérieur a certain montant requis

Une fois la preuve est vérifiée, il peut retirer le montant qui devient public mais le solde reste confidentiel. Cette preuve peut être utilisée pour démontrer la capacité du DAO à participer à des activités telles que des enchères scellées, le gestionnaire peut retirer les fonds du DAO en utilisant son autorisation spécifique. Le montant retiré devient public, mais le solde restant du DAO demeure confidentiel.

Nous décrivons ces étapes en détail dans les trois sous-sections suivantes.

Initialisation du contrat :

Lorsque le contrat principal est déployé pour la première fois, il est initialisé avec deux arbres de Merkle vide gestionnaire et contributeur, d'une profondeur donnée d (par exemple d=30). Un arbre de Merkle se présente comme une structure de données hiérarchique, où chaque nœud interne représente le hachage cryptographique des nœuds enfants. Dans le cas d'un arbre de Merkle vide, toutes les feuilles sont à zéro.

Le contrat principal stocke deux informations essentielles :

- Le hachage racine : Ce hachage représente l'état initial du contrat, étant le résultat du hachage cryptographique de la racine des arbres de Merkle vides.
- Le compteur "next" : Initialement à zéro, il indique l'emplacement de la prochaine feuille vide dans l'arbre. Ce compteur garantit un placement ordonné des dépôts dans l'arbre.
- Gestion des arbres de Merkle : Lorsqu'un dépôt est effectué dans le contrat principal, celui-ci assigne l'emplacement "next" à cette feuille, puis met à jour le compteur "next" pour pointer vers la prochaine feuille vide disponible. Tout utilisateur peut effectuer un dépôt, qui est associé à un DAO spécifique géré sur la plateforme.

Initialisation d'un nouveau contrat :

Lorsque le compteur "next" atteint $2d-1$ (où d est la profondeur de l'arbre), toutes les feuilles de l'arbre sont occupées. À ce stade, un nouveau contrat doit être initié pour continuer à recevoir les dépôts. Le processus d'initialisation du nouveau contrat est similaire à l'étape d'initialisation décrite précédemment.

L'initialisation du contrat dans notre protocole assure un stockage ordonné et efficace des dépôts dans les arbres de Merkle. Cela préserve la confidentialité des contributeurs en ne révélant pas l'emplacement spécifique de leurs dépôts dans l'arbre.

Étape 1 : Création d'un DAO

Le protocole utilise un groupe cyclique fini E d'ordre premier q avec un générateur $G \in$ E. Ce groupe est construit sur la courbe elliptique Grumpkin [19], conçue pour générer efficacement des preuves SNARK(Zero knowledge Proof) sur le réseau Ethereum. La courbe est définie sur un champ premier $\mathbb{F}r$, où r est la taille de la courbe alt_bn128 utilisée par la précompilation des pairages Ethereum [27]. Cette configuration permet d'effectuer efficacement les opérations arithmétiques sur $\mathbb{F}r$ à l'aide d'un prouveur SNARK.

Pour créer un DAO, le gestionnaire procède de la manière suivante :

- Il sélectionne aléatoirement un nombre α dans l'ensemble $\mathbb{Z}q$. Ce choix aléatoire est essentiel pour garantir la sécurité du système.
- Le gestionnaire calcule la clé publique Schnorr PK en effectuant l'opération PK ← α . G, où α est la clé secrète choisie aléatoirement et G est le générateur du groupe cyclique fini $\mathbb{E}$.
- La clé publique PK est ensuite publiée sur le site du DAO, rendue ainsi accessible au public pour permettre aux participants d'interagir avec le DAO.
- Enfin, le gestionnaire garde la clé secrète α en sécurité. Cette clé doit rester confidentielle et ne pas être divulguée, car elle lui permet de prouver son identité lors de l'exécution de certaines opérations.

Il est crucial de souligner que cette étape de création du DAO ne nécessite aucune interaction avec la chaîne Ethereum, ce qui signifie qu'aucune transaction n'est requise et qu'aucun frais de gaz n'est à payer. Cela simplifie l'expérience utilisateur et la rend plus économique lors de la création d'un DAO sur la plateforme PDAO.

Etape 2: un membre envoie des fonds au DAO:

Un client souhaite envoyer des fonds au DAO:

Le montant des fonds envoyés au DAO doit être 100 DAI par exemple.

Le client obtient la clé publique PK du DAO à partir du site web du DAO et effectue les opérations suivantes :

- . Échantillonner un nombre aléatoire ρ dans ℤq.
- . Calculer ce que nous appelons une feuille, qui est une paire :

$$L = (\rho \cdot G, \rho \cdot PK) \in \mathbb{E}\mathbf{2} ;$$

- . Calculer la preuve de Merkle stockée dans son état pour calculer la racine de l'arbre de Merkle gestionnaire. Ensuite, le contrat vérifie que la preuve de Merkle fournie pour la position suivante est cohérente avec la racine de l'arbre mise à jour.
- Le client sélectionne aléatoirement un nombre α client dans Zq.
- Le client calcul sa propre clé publique schnorr PK client ← α client . G
- Ensuite, le client garde sa clé secrète α client en sécurité car elle va lui servir à reprendre les fonds qu'il a envoyé au contrat.
- Après l'étape de génération de la paire de clé du client, le client va calculer une feuille pour le second arbre de Merkle : (Merkle client).

$$L_{clent} = (P.G, P.PK\ client) \in E2$$

- Appeler la fonction deposit ( ) du contrat principale avec les arguments L , L_{clent} en envoyant 100 DAI dans la transaction.
- . Calculer la preuve de Merkle pour les prochains emplacements, soit la position suivante dans les deux arbres, celui correspondant au gestionnaire des DAO est celui correspondant aux clients.

Le contrat utilise L et L' et les preuves de Merkle stockées dans son état afin de calculer les deux racines de Merkle mise à jour.

Le contrat principal accepte les fonds et enregistrer les feuille étendue :

$$L^{'} = (P.G, P.PK, 100, n)$$

$$L'_{clent} = (P.G, P.PK_{client}, 100, n)$$

"n" représente la position dans laquelle les deux feuilles sont insérées. Rappelons que les deux arbres de Merkle dans notre position sont de la même taille et contiennent le même nombre de feuilles non nulles.

Le contrat incrémente next et calcule les deux racines de Merkle des deux arbres après l'ajout des deux feuilles $L^{'}$ et L'_{client} .

Comme les feuilles $L^{'}$ et L'_{clent} sont publiées sur la chaîne et sont visibles par le monde entier, il est impératif que L' ne révèle rien sur l'identité du DAO qui a reçu les fonds. En particulier, rien sur PK ne doit être révélé. Cela découle facilement de la difficulté de l'assomption de décision Diffie-Hellman[28] dans le groupe E.

Étape 3 : Le gestionnaire du DAO retire des fonds

Lorsque le gestionnaire du DAO souhaite retirer w ETH du DAO, il doit le faire en prouvant que le DAO était le destinataire prévu d'au moins cette quantité d'ETH. De plus, ces fonds ne doivent être retirés qu'une seule fois. Tout cela doit être fait sans révéler le solde du DAO au contrat.

Nous y parviendrons en utilisant un SNARK.

Tout d'abord, comment le gestionnaire du DAO connaît-il même le solde des fonds envoyés au DAO ? Pour ce faire, le gestionnaire du DAO surveillera tous les dépôts envoyés au contrat principal. Pour chaque feuille observée L' = (P, Q, v, n), le gestionnaire du DAO utilise sa clé secrète α pour vérifier si : α . P = Q.

Si tel est le cas, le gestionnaire du DAO apprend que ce dépôt est destiné à son DAO et enregistre que v ETH ont été ajoutés au solde du DAO. Sinon, ce dépôt est destiné à un autre

DAO.

Le protocole de retrait :

Supposons maintenant que le gestionnaire du DAO souhaite retirer w ETH du DAO. Nous exigeons qu'il existe un sous-ensemble de feuilles $\{L'i=(Pi ,Qi ,vi ,ni)\}\ell i=1$ qui appartiennent au DAO tel que $v1+ \ldots+v\ell=w$.

Un protocole de retrait simple consiste à ce que le gestionnaire du DAO appelle la fonction withdraw() sur le contrat principal en lui donnant l'argument :

$(n1,\ldots,n\ell,\pi)$, où

π : représente une preuve SNARK que le gestionnaire DAO possède, avec un témoin : (α , $P1,Q1,v1, \ldots , P\ell,Q\ell,v\ell$)

Il est impératif que les conditions suivantes doivent être respectées :

1. Pour $i=1,\ldots,\ell$ la feuille (Pi,Qi,vi,ni) est dans l'arbre de Merkle T

2. Pour $i=1,\ldots,\ell$ on a $Qi=\alpha\cdot Pi$

3. $v1+\cdots+v\ell=w$.

Le contrat principal vérifie la preuve SNARK π et vérifie que les feuilles $n1,\ldots,n\ell\in[2^d]$ n'ont pas encore été dépensées. Si tel est le cas, le contrat principal envoie w ETH au gestionnaire DAO, à l'origine de la demande de retrait. Il ajoute ensuite $n1,\ldots,n\ell$ à sa liste de feuilles épuisées nullifier afin que ces feuilles ne puissent plus être dépensées.

Puisqu'un SNARK nécessite une entrée de taille fixe, nous pouvons définir $\ell=100$, afin que le gestionnaire DAO puisse retirer un lot d'une centaine de feuilles à la fois.

Etape 3 bis:

L'étape qui va suivre constitue notre contribution dans ce mémoire. Lors de l'explication de l'approche proposée par Boneh et Dunaif, nous pouvons remarquer que cette approche ne donnait pas la possibilité au client de reprendre les fonds qu'il avait déposé dans le cas où il le souhaitait

Grâce à l'approche que nous proposons , ceci devient possible;

Protocol de retrait Bis (le client retire l'argent qu'il a versé):

.Nous exigeons qu'il existe une feuille $L_{i\,client}$ dans l'arbre client. $L_{i\,client} = (P_i, Q_i V_i, n_i)$

Le client va faire appelle à la fonction withdraw_client ( ) par le contrat principal en lui donnant l'argument : (n,$\boldsymbol{\pi}$) où :

$\boldsymbol{\pi}$: représente une preuve SNARK que le client possède , avec un témoin (α , P , Q , V)

Il est impératif que les conditions suivantes doivent être respectées:

- La feuille (P, Q , V , n) est dans l'arbre client Q= α_{client}.p
- Le contrat principal vérifie la preuve SNARK $\boldsymbol{\pi}$ et vérifie que la feuille n'a pas déjà été retirée par le client , et surtout qu'elle n'a pas déjà été dépensée par le gestionnaire de DAO à qui elle était destinée . Si tel est le cas , le contrat envoie le montant V au client et inscrit la feuille de l'arbre gestionnaire correspondante dans la liste Nullifier des feuilles dépensées.

3.6 Conception de notre solution :

3.6.1 Le but de notre produit :

Pour assurer l'adoption généralisée des DAO, en particulier par les entreprises, il est essentiel que les actifs des DAO restent privés. Cela signifie que seuls les gestionnaires du DAO

devraient connaître le total des actifs gérés par le DAO, tandis que cette information est cachée au reste du monde.

Il existe plusieurs cas d'utilisation où la confidentialité des actifs est cruciale :

- DAO d'investissement : Un DAO gérant un fonds d'investissement peut vouloir que la taille de son fonds reste privée.
- Entreprises : Une entreprise qui utilise un DAO pour gérer une partie de ses activités peut vouloir garder privées certaines informations financières, telles que les chiffres de ventes actuels.
- Fonds participant à des enchères : Un fonds qui participe à des enchères, par exemple pour l'achat de biens ou de services, a besoin que son trésor reste privé pour des raisons de stratégie d'enchères.

Ce projet vise à résoudre les problèmes liés à l'exposition d'informations sensibles sur les actifs, en particulier ceux causés par le concept de trésorerie publique. Il propose un mixeur de fonds sous la forme d'une trésorerie collective, où plusieurs DAO peuvent mélanger les fonds reçus de leurs contributeurs au sein d'un seul contrat de trésorerie. L'objectif est d'introduire un anonymat sur le montant des fonds reçus par ces DAO, préservant ainsi la confidentialité des informations financières et encourageant une adoption plus large et plus confiante des DAO. Le second objectif est de permettre aux contributeurs de la DAO de se faire rembourser leurs fonds dans le cas où ils le désirent et où c’est encore possible (les fonds n’ont pas été dépensés).

3.7 La portée du produit :

Notre système utilise un protocole visant à assurer la confidentialité des contributions, la protection des fonds et la gestion privée des actifs au sein des DAO (Organisations Autonomes Décentralisées). En exploitant des technologies telles que le mixing, les SNARKs, les arbres de Merkle et le protocole Tornado Cash, notre système crée un environnement sécurisé et confidentiel.

Son objectif principal est de permettre aux utilisateurs de créer et de gérer des trésoreries DAO tout en préservant la confidentialité de leurs informations sensibles. Il répond aux préoccupations liées à la divulgation des données qui pourraient compromettre le bon fonctionnement de ces organisations, tout en assurant la protection des fonds et en offrant une transparence sélective au sein des DAO. Les objectifs clés du protocole incluent :

- Confidentialité des soldes : Les fonds envoyés à un DAO ne révèlent pas à quel DAO ils sont destinés. Bien qu'un observateur puisse voir le montant total soumis à tous les DAOs sur la plateforme en lisant le solde du contrat maître, il ne peut rien apprendre sur le solde de chaque DAO, autre que cette borne supérieure.
- Calcul du solde par le gestionnaire du DAO : Le gestionnaire du DAO peut calculer le solde actuel des fonds envoyés à son DAO. De plus, il doit pouvoir prouver à un tiers que le solde du DAO est supérieur à un certain montant.
- Confidentialité des retraits : Lorsque le gestionnaire du DAO retire des fonds, le montant retiré est public, mais le solde restant du DAO reste caché.
- Possibilité de rétractation des fonds pour les contributeurs : Les contributeurs doivent pouvoir retirer leurs fonds du DAO.

3.8 Perspective :

Notre protocole est similaire à celui utilisé par Tornado.cash Classic, un mélangeur d'anonymat, mais les propriétés de confidentialité des deux protocoles présentent des différences significatives. Dans Tornado.cash, l'expéditeur et le destinataire sont la même personne : un utilisateur dépose des fonds dans le contrat Tornado et retire plus tard ces mêmes fonds. Dans notre système, en revanche, l'expéditeur et le destinataire sont distincts. Lorsque le gestionnaire du DAO retire des fonds, un observateur pourra identifier à quel contributeur ces fonds appartenaient. De plus, si un contributeur souhaite retirer ses propres fonds, tous les observateurs pourront voir qui a effectué le retrait et connaître le montant retiré.

3.9 Caractéristiques du Système

Notre système se distingue par deux fonctionnalités principales :

1. Dépôt des fonds : Cette fonctionnalité permet aux utilisateurs de déposer des actifs en tant que contributions dans le contrat principal, en les associant cryptographiquement au DAO de leur choix, tout en préservant la confidentialité du DAO concerné.

2. Retrait des fonds: Cette fonctionnalité permet à un type spécifique d'utilisateur, les gestionnaires de DAO, de retirer des actifs du contrat principal. Ils doivent prouver qu'ils sont bien des gestionnaires d'un DAO enregistré dans le contrat, sans révéler l'identité du DAO. Ces gestionnaires sont uniquement autorisés à retirer des fonds associés à leur propre DAO.
Une seconde version de cette fonctionnalité va permettre à un contributeur de reprendre les fonds qu’il a donné en donnant une preuve cryptographique que ses fonds lui appartiennent.

3.10 Diagramme de Cas d'utilisation :

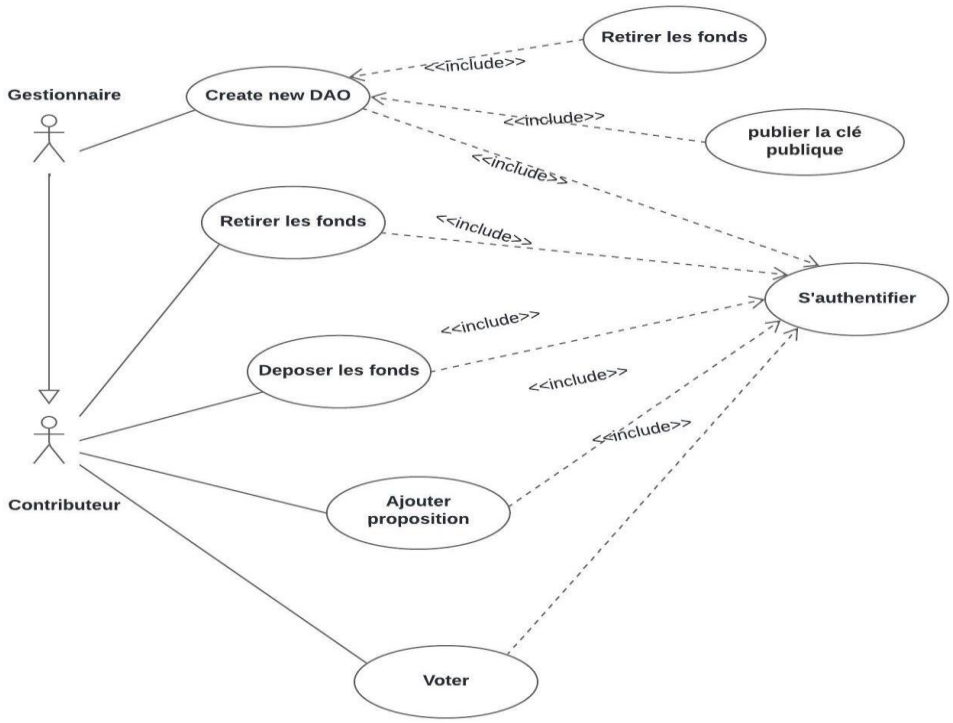

Figure 16: Diagramme de cas utilisation générale

3.10.1 Les Acteurs :

1. Gestionnaire :

- Création du DAO (hors blockchain)
- Publier la clé du DAO
- Retraits de fonds

2. Contributeur :

- Dépôts de fonds.
- Retirer les fonds
- Ajouter proposition
- Voter

3.11 Diagramme de Classe :

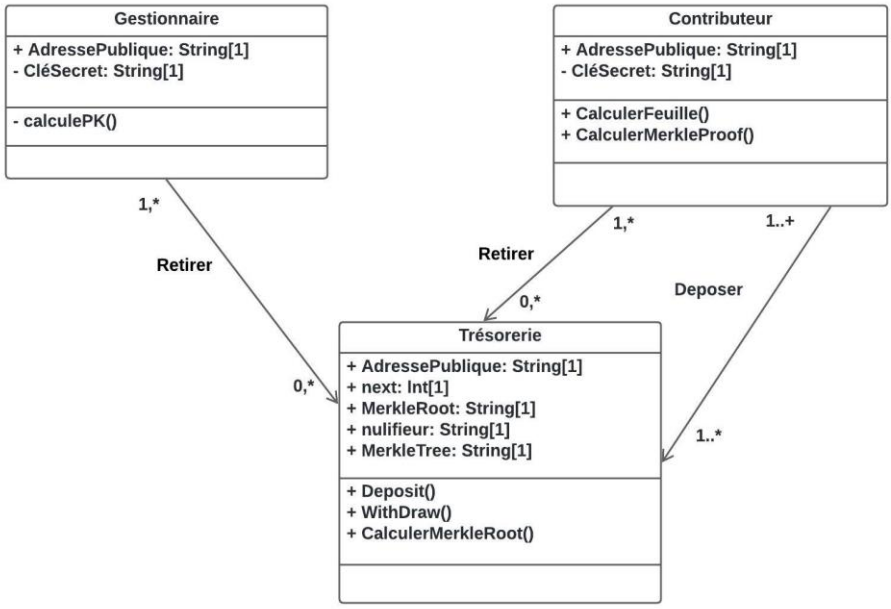

Figure 17 : Diagramme de classe

3.11.1 Description des Classes :

Gestionnaire :

Cette classe représente le gestionnaire DAO, responsable de la création et de la gestion du DAO. Elle contient les méthodes suivantes :

- calculePK() : Cette méthode calcule la clé publique associée au DAO.

Contributeur :

Cette classe représente l'utilisateur final souhaitant contribuer avec des fonds au DAO. Elle contient les méthodes suivantes :

- calculeFeuilleL() : Cette méthode calcule la feuille représentant cryptographiquement le DAO auquel ces fonds seront attribués.
- calculeMerkleProof() : Cette méthode génère la série de hachages allant de la racine à la feuille finale où le prochain nœud sera enregistré.

Trésorerie :

Cette classe représente le contrat intelligent responsable de vérifier les preuves fournies par le gestionnaire DAO lors d'un retrait de fonds. Elle contient les méthodes suivantes :

- calculeMerkleRoot() : Cette méthode recalcule la preuve fournie par le contributeur.
- deposit(): Permet de recevoir des contributions.
- withdraw() : Permet de retirer des fonds du contrat.

3.12 Diagramme de séquence d'authentification :

Ce diagramme présente le processus d'authentification impliquant la connexion du portefeuille (wallet) utilisateurs du système la figure ci-dessous montre comment cela se déroule.

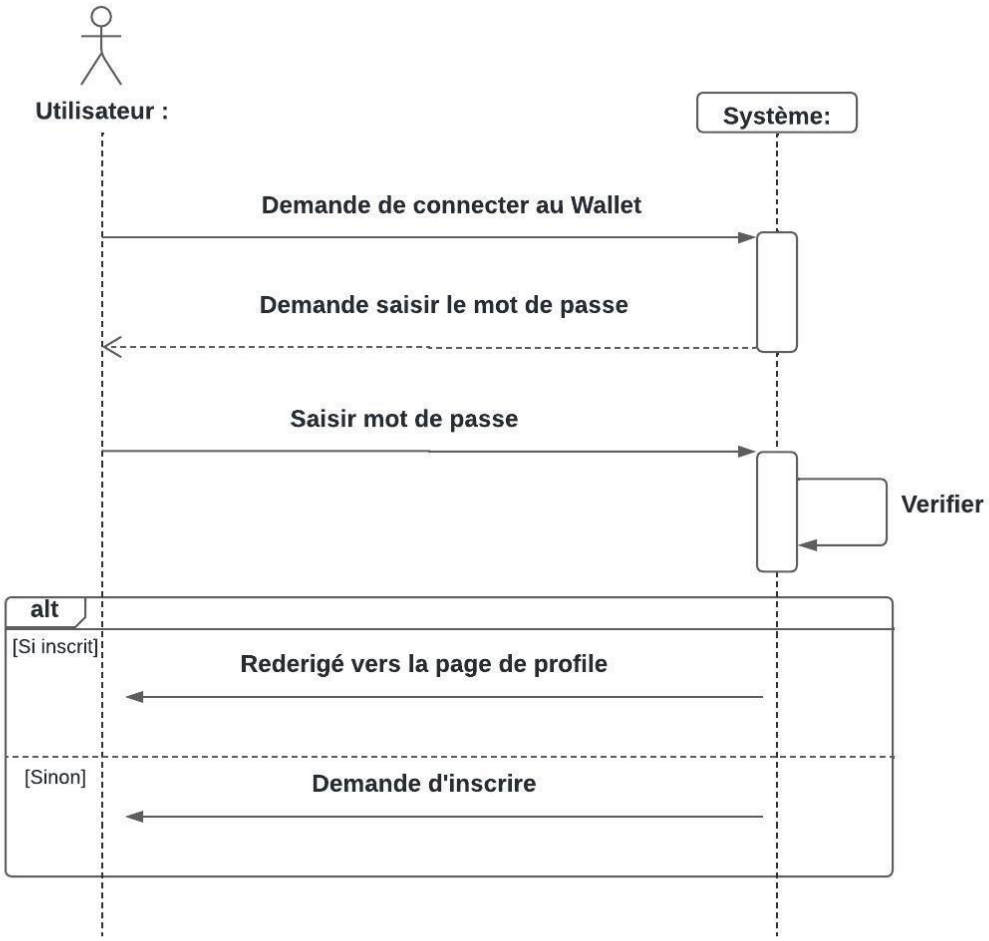

Figure 18 : Diagramme de séquence d'authentification

3.13 Diagramme de séquence d'ajouter proposition :

La figure ci-dessous représente l'opération d'ajouter une proposition

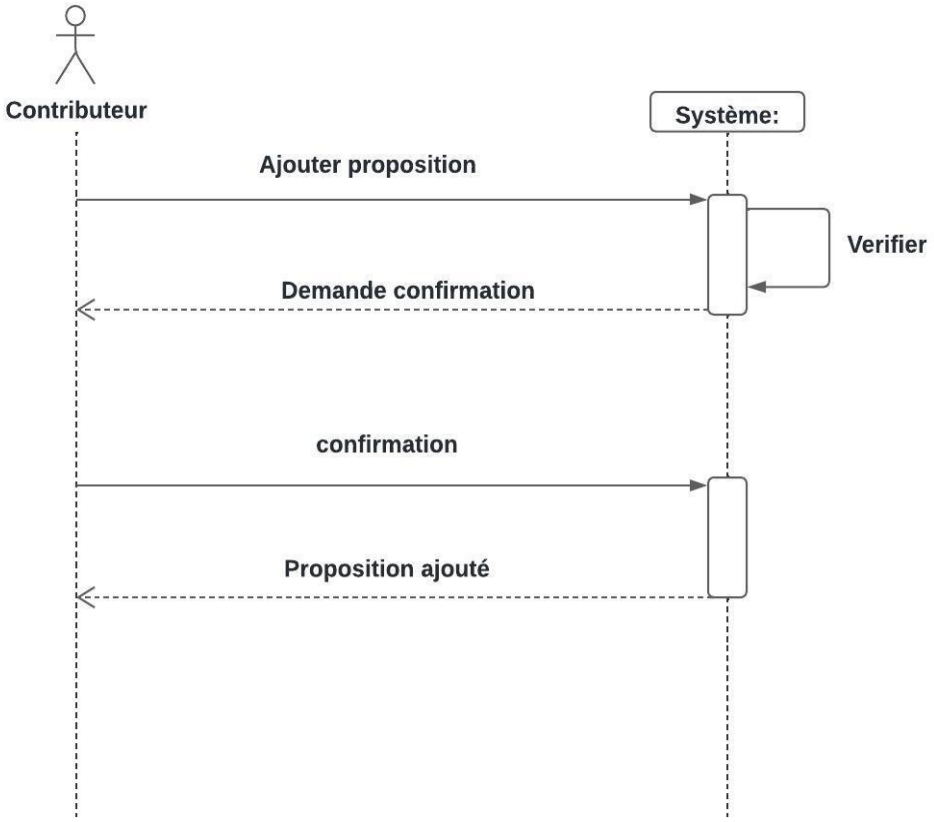

Figure 19 : Diagramme de séquence d'ajouter proposition

3.14 Diagramme de séquence de l'opération du vote :

La figure ci-dessous représente l'opération du vote

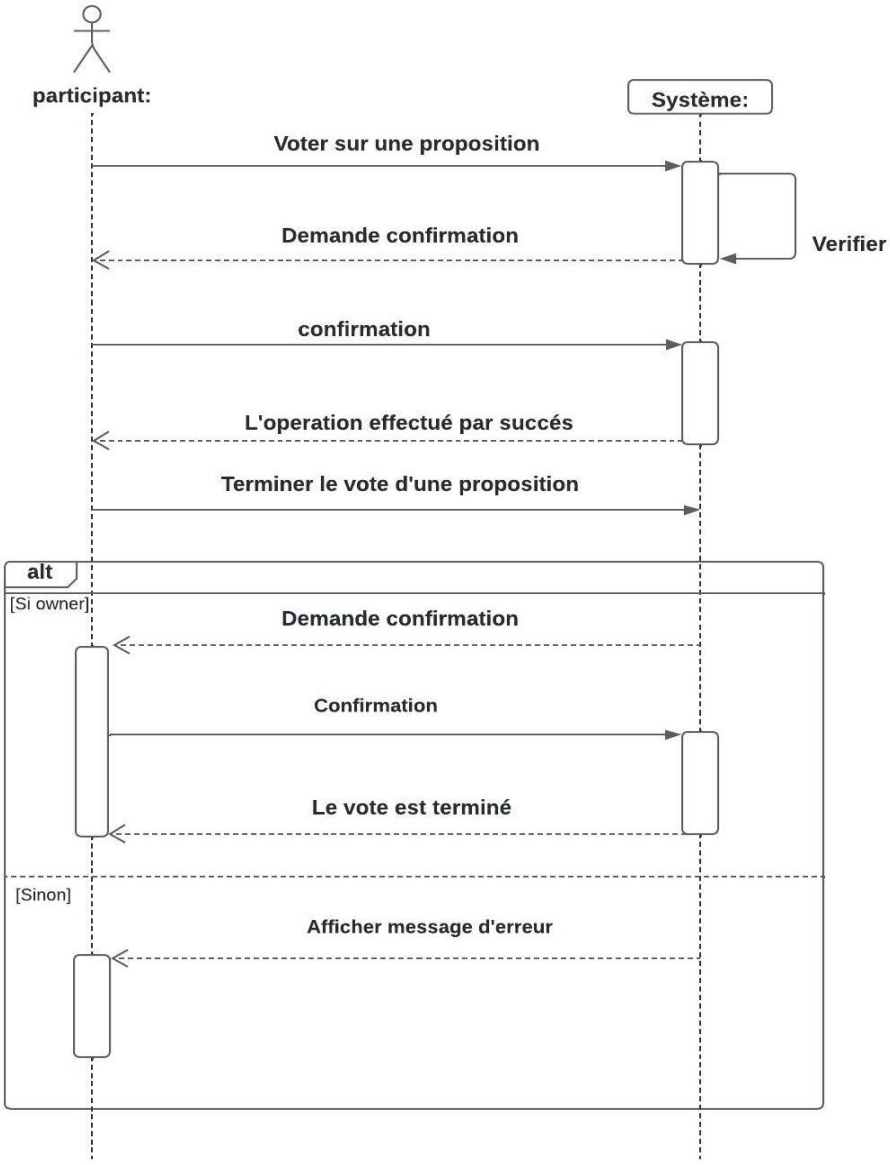

Figure 20 : Diagramme de séquence de l'opération du vote

3.15 Diagramme de séquence du cas Publier la clé du Dao :

Cette étape est essentiel pour la création du dao afin d'obtenir les fonds

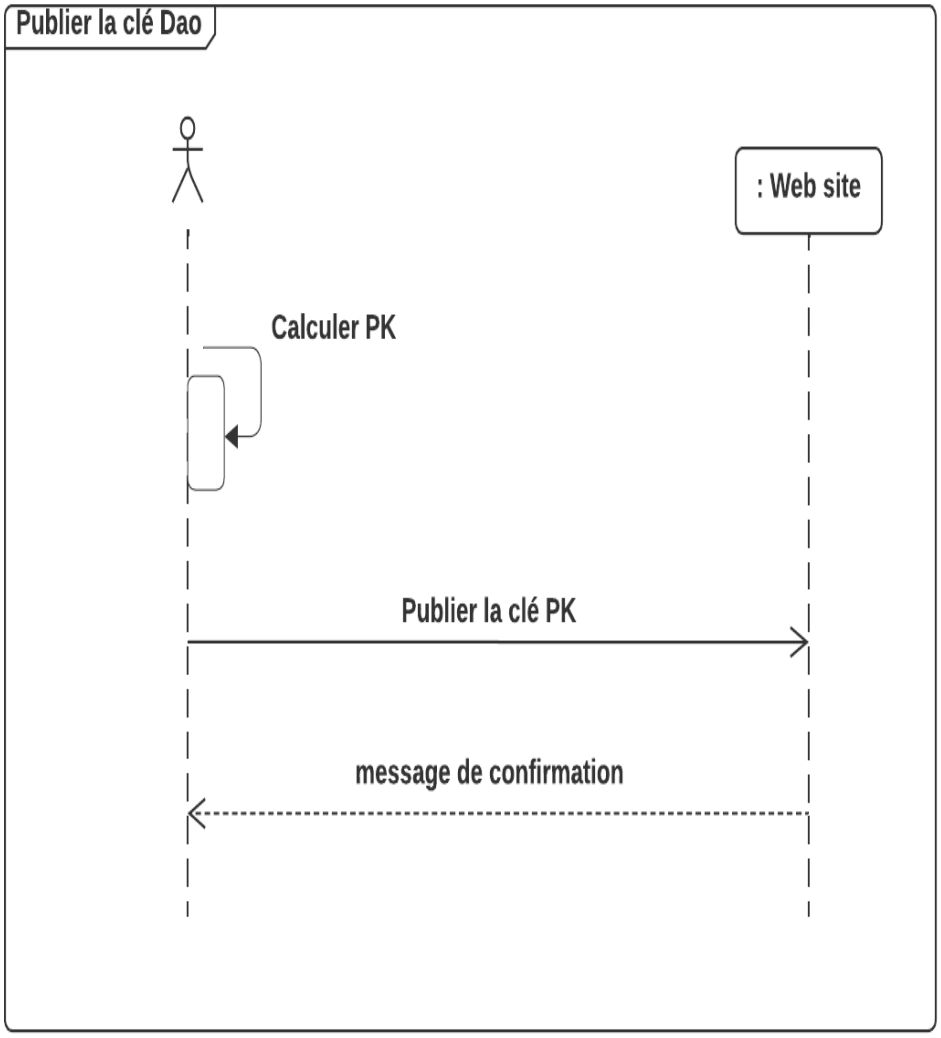

Figure 21: Diagramme de séquence du cas Publier la clé du Dao.

3.16 Diagramme de séquence de Déposer des fonds :

Le diagramme présenté ci dessous lorsque le contributeur veut contribuer au Dao en faisant un Dépôt au contrat**:**

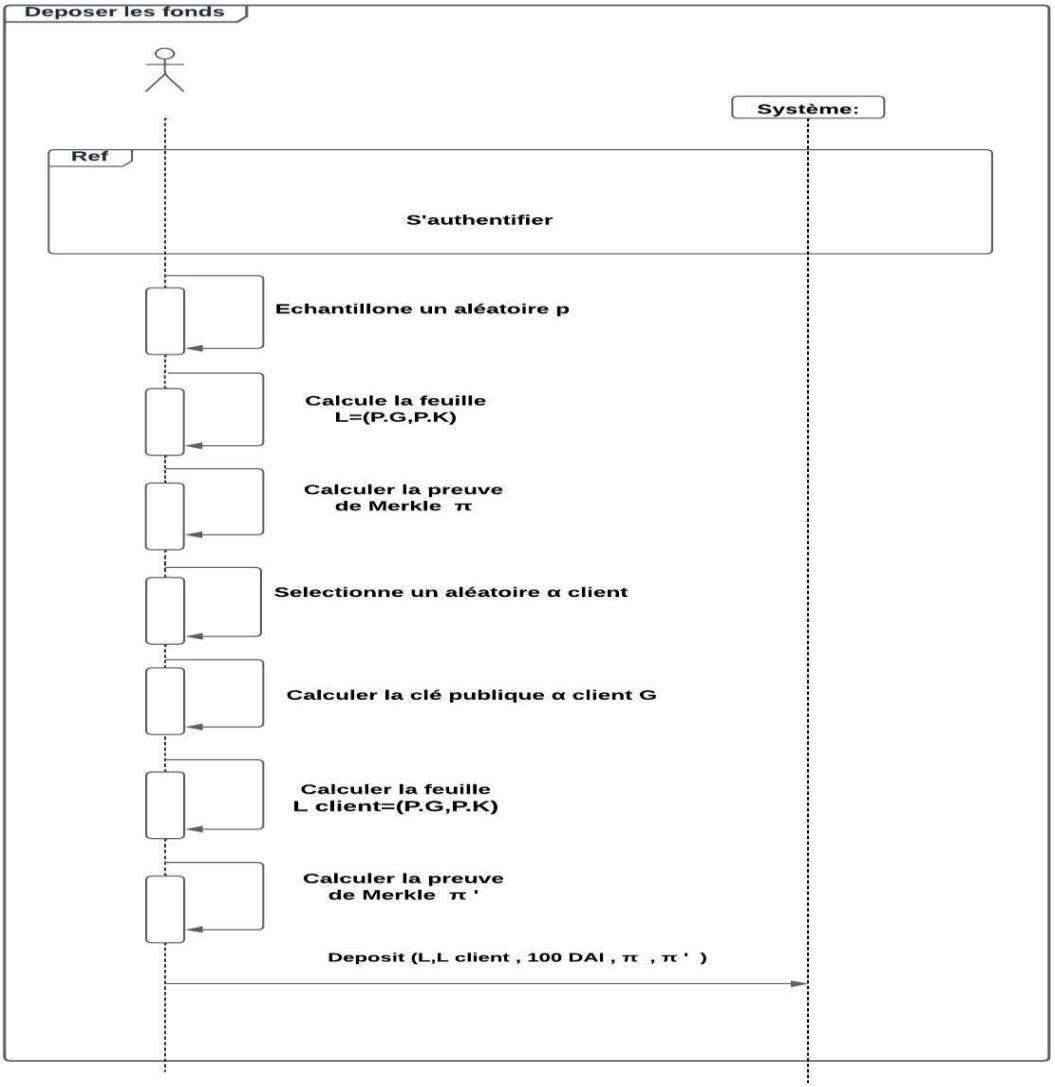

Figure 22: Diagramme de séquence de Déposer des fonds.

3.17 Diagramme de séquence de Retirer des fonds pour le gestionnaire:

Le diagramme ci-dessous présente lorsque le gestionnaire veut retirer les fonds

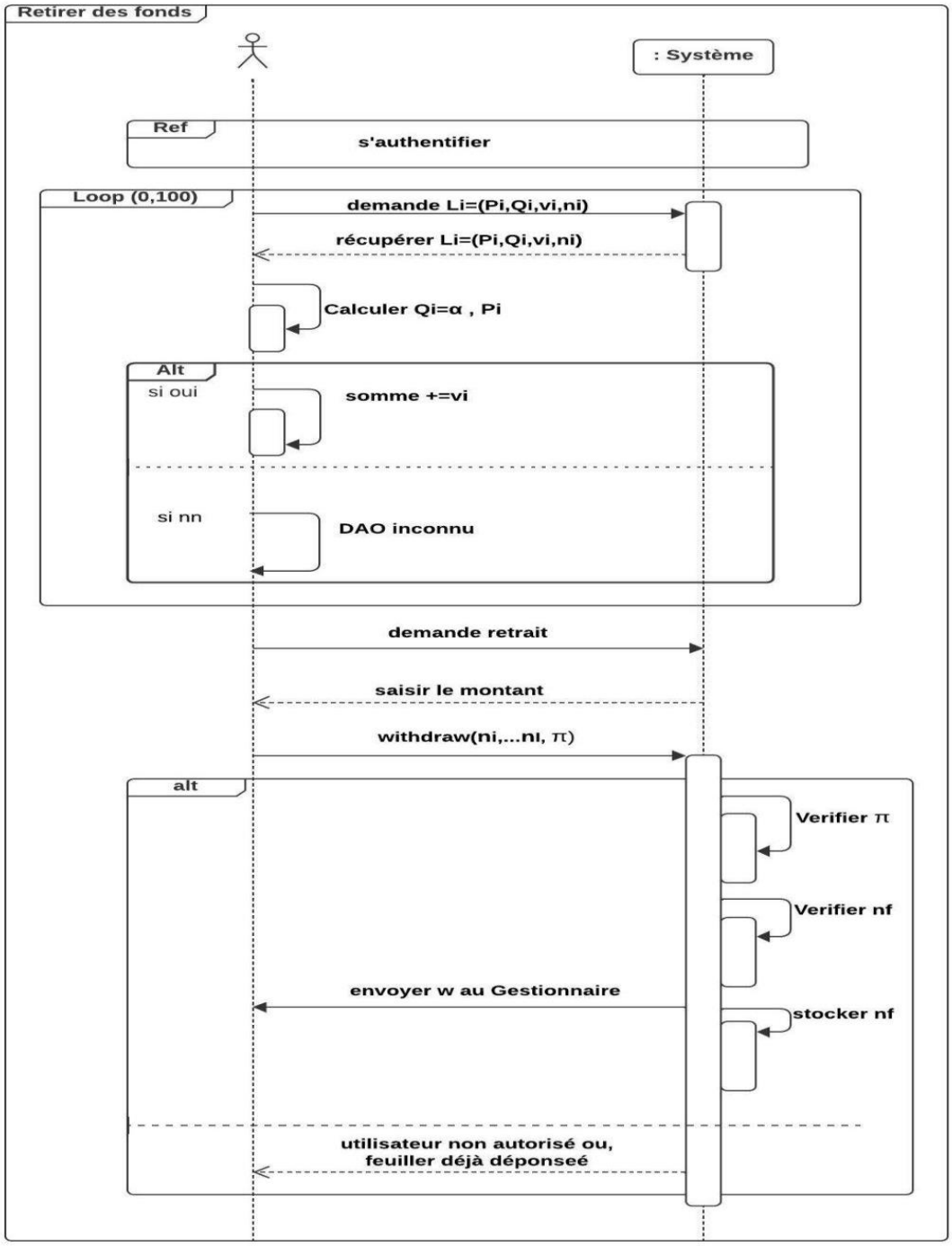

Figure 23: Diagramme de séquence de Retirer des fonds pour le gestionnaire.

3.18 Diagramme séquence de Retirer les fonds pour le contributeur :

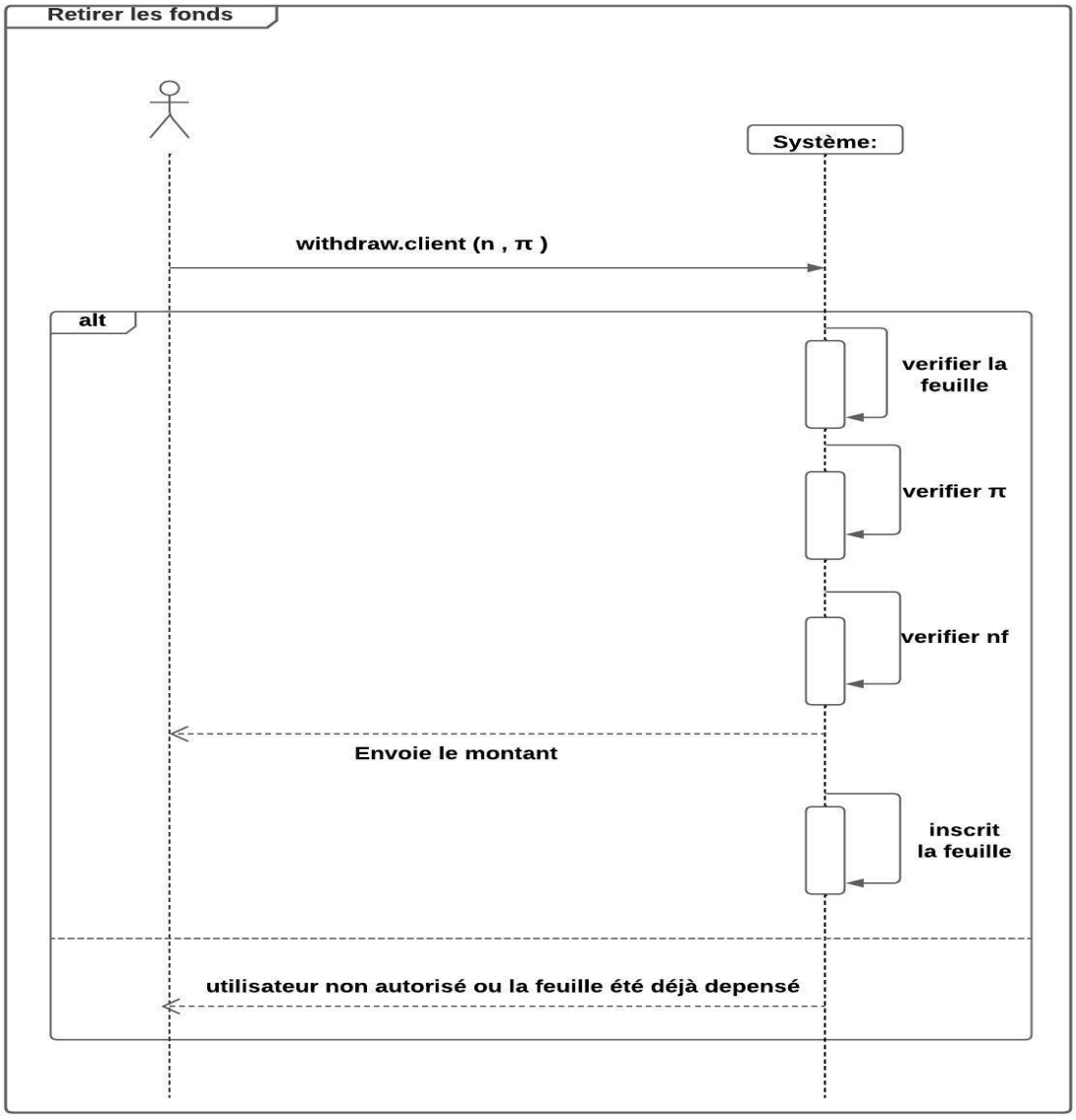

Figure 24 : diagramme de séquence pour retirer les fonds pour le contributeur

Discussion de notre proposition :

La solution que nous proposons remplit un certain nombre d'objectifs de confidentialité, cependant, elle n'est pas parfaite. La confidentialité des contributions est garantie. le solde du DAO reste confidentiel et ne sera révélé que lorsque les fonds voudront être dépensés. La solution dont nous nous sommes inspirés ne fournissait pas au contributeur la possibilité de reprendre les fonds qu'il avait initialement investis. Ceci est un frein considérable pour l'adopter, sachant que les DAOs ont pour habitude de permettre la reprise des fonds par leur propriétaire. Notre solution résout ce problème et améliore ainsi la proposition initiale.

3.19 Conclusion :

Au cours de ce chapitre, nous avons approfondi notre discussion sur un protocole similaire à celui utilisé par le célèbre mélangeur d'anonymat appelé Tornado.cash sur le réseau Ethereum. Nous avons mis en lumière les problèmes liés à la transparence financière des organisations décentralisées autonomes (DAO) dans une trésorerie publique, où toutes les transactions sont visibles, ainsi que les défis spécifiques liés à la participation des DAO à des enchères scellées, en prenant comme exemple la ConstitutionDAO.

Par la suite, nous avons défini les objectifs et analysé en détail le cycle de vie d'une gestion de trésorerie confidentielle pour un DAO proposé par Dan Boneh et griffin dunaif. Ce cycle comprend trois étapes clés : la création du DAO, le dépôt de fonds et le retrait de ces fonds. Ensuite, nous avons proposé notre propre solution, mettant en évidence les mesures prises pour garantir la confidentialité, protéger les fonds et permettre une gestion privée des actifs par le gestionnaire, tout en donnant la possibilité au contributeur de récupérer les fonds qu'il avait initialement investi.

Chapitre 4

Implémentation

4.1 Introduction :

Dans ce chapitre, nous nous concentrerons sur la réalisation et le déploiement de notre application dédiée à la gestion des trésoreries pour les DAO, qui souhaitent utiliser notre système pour gérer leurs fonds. Nous présenterons les outils de développement que nous avons adoptés, à savoir la Blockchain Ganache pour la gestion des transactions, le langage Solidity pour la création des contrats intelligents côté backend, ainsi que le framework React et JavaScript pour le développement frontend. En outre, nous intégrerons la base de données MySQL via le serveur XAMPP pour gérer les processus d'inscription et d'authentification des utilisateurs dans le backend.

4.2 Outils de développement :

4.2.1 La blockchain ganache :

Ganache est une blockchain personnelle conçue pour le développement rapide d'applications distribuées Ethereum offrant un environnement de développement local sécurisé Vous pouvez utiliser Ganache tout au long du cycle de développement, ce qui vous permet de concevoir, déployer et tester vos dApps (applications décentralisées) dans un environnement sûr et déterministe.

Ganache se décline en deux versions : une interface utilisateur (UI) et une interface en ligne de commande (CLI). Ganache UI est une application de bureau prenant en charge la technologie Ethereum[29].

4.2.2 Langage Solidity :

Solidity est un langage de programmation orienté objet, conçu spécifiquement pour écrire des smart contracts sur la blockchain Ethereum. Il est influencé par des langages tels que JavaScript, ce qui facilite la transition pour les développeurs familiarisés avec ce type de langage. Solidity est un langage de haut niveau qui permet de définir des structures de données complexes, des fonctions et des interactions avec d'autres Smart contracts[30].

4.2.3 Remix Ethereum :

Remix Ethereum est un environnement de développement intégré (IDE) qui offre aux développeurs plusieurs fonctionnalités essentielles. Il leur permet de rédiger, tester, déboguer et déployer des contrats intelligents (smart contracts) sur la blockchain Ethereum. De plus, Remix propose des interfaces graphiques intuitives qui facilitent l'écriture, le test et le déploiement du code, permettant ainsi de vérifier la fonctionnalité des contrats intelligents [31].

4.2.4 JavaScript :

JavaScript est un langage de programmation utilisé pour ajouter des fonctionnalités interactives et dynamiques aux pages web. Il est essentiel lorsque vous avez besoin de créer des éléments tels que des mises à jour de contenu en temps réel, des cartes interactives, des animations, des menus déroulants, et bien plus encore. JavaScript est considéré comme la troisième couche des technologies web standard, complétant ainsi HTML et CSS [32]**.**

4.2.5 React js :

React est une bibliothèque JavaScript déclarative, efficace et flexible pour construire des (UI). Elle vous permet de composer des UI complexes à partir de petits morceaux de code isolés appelés « composants »[33].

4.2.6 MetaMask :

C’est une extension de navigateur Web et une application mobile qui vous permet de gérer vos clés privées Ethereum. Ce faisant, il sert de portefeuille pour Ether et d'autres jetons, et vous permet d'interagir avec des applications décentralisées ou dapps. Contrairement à certains portefeuilles, MetaMask ne conserve aucune information sur vous, c'est-à-dire que votre adresse e-mail, votre mot de passe et votre phrase de récupération secrète ou d'autres clés privées ne sont pas conservés. Vous conservez tout le pouvoir sur votre identité crypto [34].

4.2.7 Web3.js :

Est une bibliothèque JavaScript qui permet aux développeurs d'interagir avec la blockchain Ethereum et de construire des applications décentralisées . Elle offre une interface simple et conviviale pour accéder aux données de la blockchain et exécuter des contrats intelligents [35].

4.2.8 MySQL :

C’est un système de gestion de base de données relationnelles (SGBDR) populaire, distribué sous une double licence GPL(Licence Publique Générale) et propriétaire. Il est largement utilisé dans le monde entier, aussi bien par le grand public, principalement pour les applications web, que par les professionnels. MySQL rivalise avec des concurrents tels qu'Oracle, PostgreSQL et Microsoft SQL Server [36].

4.2.9 XAMPP Server :

Est un ensemble de logiciels qui permet de configurer facilement un serveur Web local
XAMPP est accessible à un large public car il ne nécessite pas de connaissances particulières et fonctionne sur les systèmes d'exploitation les plus courants. De plus, il est livré avec

phpMyAdmin, un outil permettant d'administrer des bases de données en utilisant MySQL[37].

4.2.10 Express js :

Est un framework d'application web côté serveur pour la création d'API RESTful avec Node.js. Il est distribué en tant que logiciel libre et open-source sous la licence MIT(Massachusetts Institute of Technology). Express est conçu pour la construction d'applications web et d'API. Il est considéré comme le framework serveur de facto pour Node.js[38].

4.2.11 Visual Studio Code (VSCode):

C'est un éditeur de code source et un environnement de développement intégré (IDE) de Microsoft. Il est open-source et cross-platform, c'est-à-dire qu'il fonctionne sur Windows, Linux et Mac. Il a été conçu pour les développeurs web, mais il prend en charge de nombreux autres langages de programmation tels que C++, C#, Python, Java, etc. Il offre de nombreuses fonctionnalités comme la coloration syntaxique, l'auto-complétion, la mise en évidence des erreurs, la navigation de code, le débogage, la gestion de versions, l'intégration avec Git, et beaucoup d'autres. Il est également extensible à l'aide d'une grande variété d'extensions développées par la communauté, permettant aux développeurs de personnaliser l'éditeur selon leurs besoins [39].

4.3 Organigrammes de l'application :

4.3.1 Interface de MetaMask :

L'utilisateur doit se connecter à la Blockchain en utilisant l'extension MetaMask, pour s'inscrire au site du DAO par son compte ce qui lui permet d'utiliser son propre portefeuille Ethereum pour signer ses transactions.

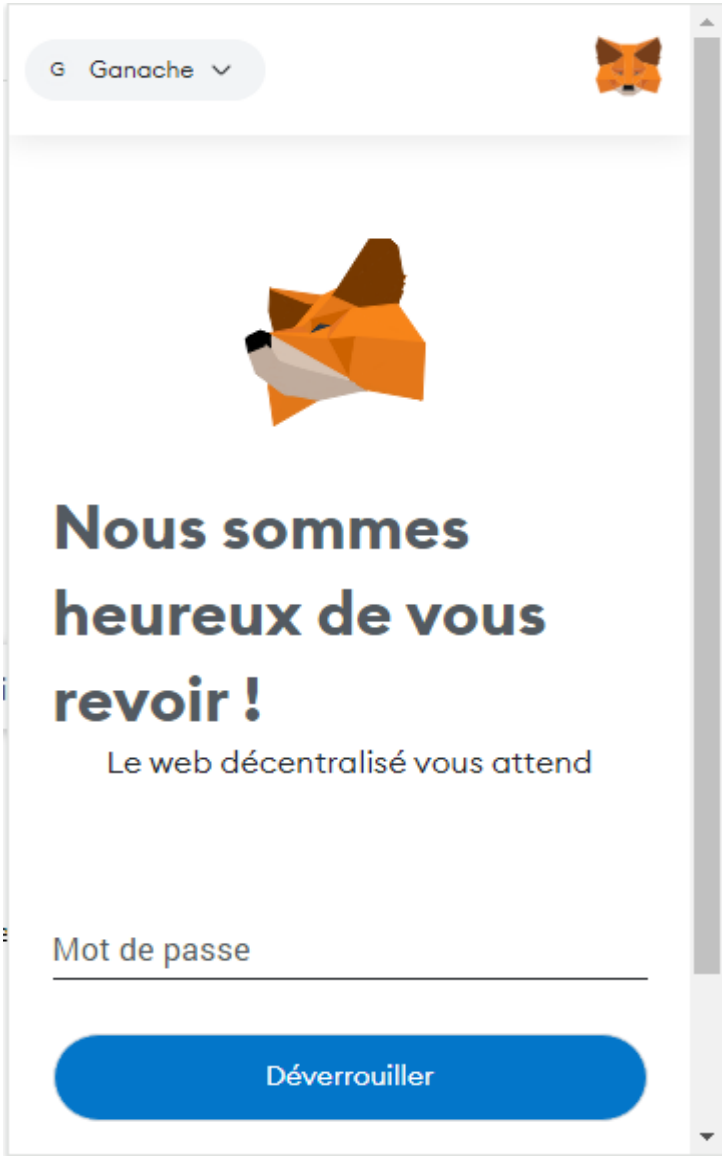

Figure 25 : Interface de connexion MetaMask.

4.3.2 Interface d'inscription :

Lorsqu'un utilisateur souhaite participer au DAO, il doit s'inscrire et remplir les informations du formulaire, en fournissant l'adresse correspondant à son portefeuille MetaMask.

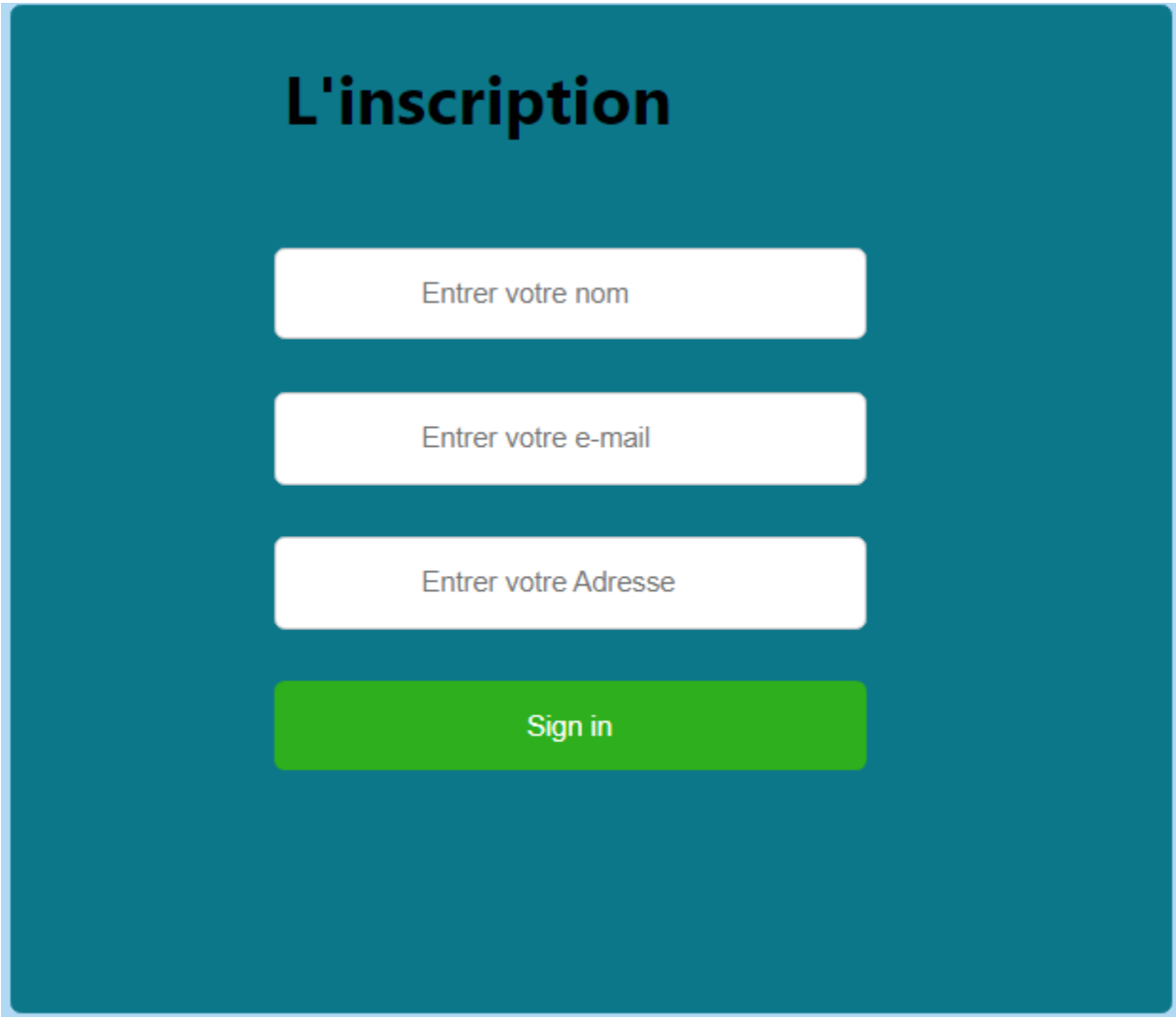

Figure 26 : Interface d'inscription

4.3.3 Interface de contribution :

Cette figure 27 permet aux contributeur de déposer des fonds dans la balance du contrat principal ainsi de retirer leurs fonds

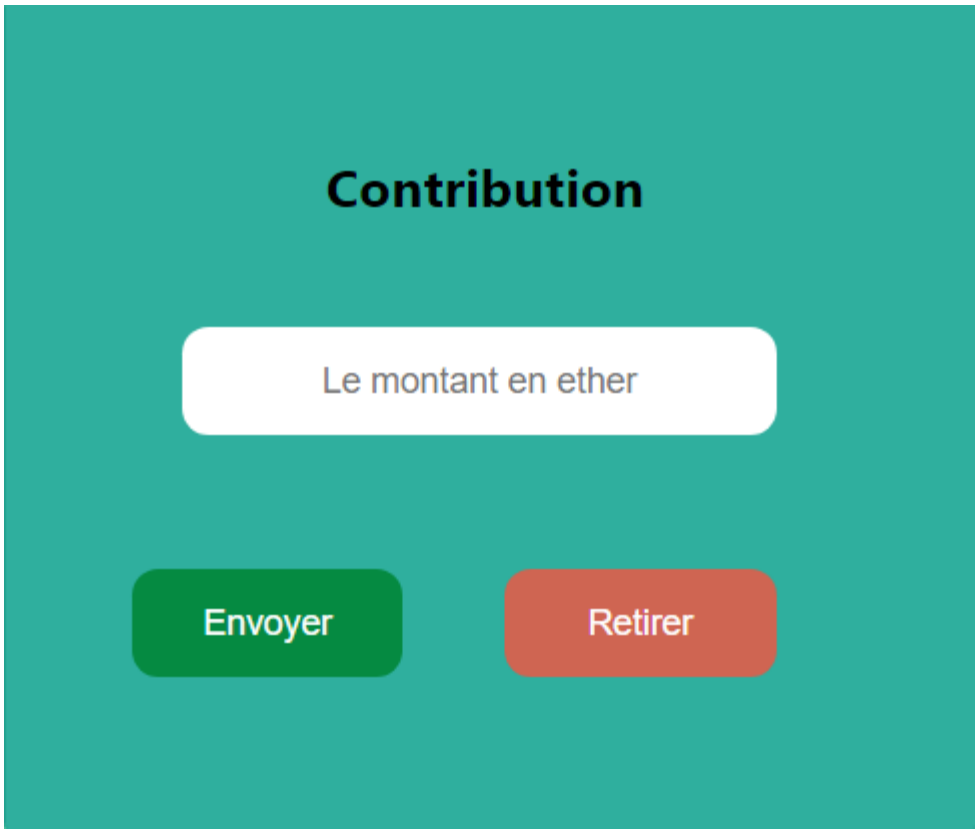

Figure 27 : Interface de contribution

4.3.4 Interface d'ajout proposition et du vote :

Les contributeur peuvent donner des propositions au DAO et voter sur ces proposition lorsqu'ils sont en cours afin d'améliorer le fonctionement du DAO

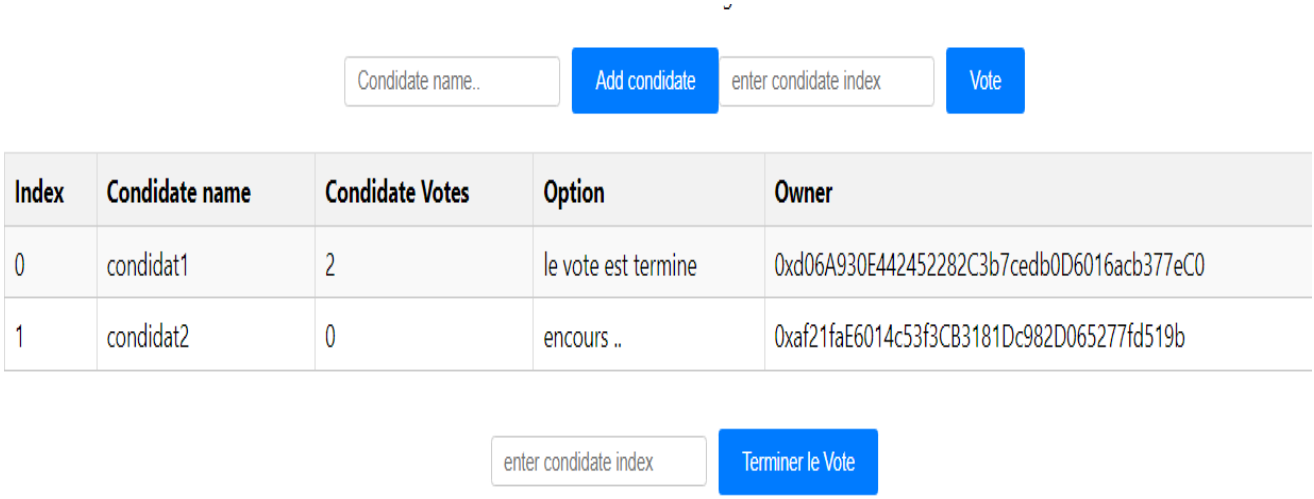

Figure 28 : Interface d'ajout proposition et du vote

4.4 Conclusion :

Dans cette dernière section, nous avons abordé la réalisation et l'implémentation de notre projet en spécifiant nos choix en matière d'outils, de langages et d'environnements de développement. Nous avons présenté notre solution basé sur tornado cash

Notre solution consiste en un Smart Contrat et une base de données en Backend et une application React Cliente en Frontend.

Conclusion générale :

La technologie blockchain, en dépit de sa jeunesse, a profondément transformé notre manière d'appréhender la sécurité et la gouvernance des échanges en ligne. Les organisations autonomes décentralisées (DAO) en sont un parfait exemple, offrant une alternative novatrice aux structures centralisées traditionnelles.

Cependant, cette transparence absolue, bien que constituant un atout majeur des DAO, pose également des défis en matière de sécurité et de confidentialité. Des solutions doivent être mises en place pour garantir l'intégrité et la protection des données au sein de ces entités décentralisées.

L'émergence des DAO marque une étape significative dans l'évolution de la gouvernance et des organisations décentralisées. Comprendre pleinement les enjeux et les opportunités offerts par ces nouvelles structures est essentiel pour appréhender pleinement le potentiel de la technologie blockchain dans divers secteurs.

Nous avons au cours de notre PFE traité une question contradictoire et passionnante : peut-on avoir de la confidentialité dans une blockchain qui est par définition transparente ? Nous avons pu apporter un début de réponse à cette question grâce à la cryptographie et notamment grâce au protocole Tornado cash, ainsi qu'au protocole que nous avons proposé. Ces deux solutions répondent toutes les deux à quelques problématiques de confidentialité dans la blockchain, mais d'autres restent ouvertes et propices à de futures recherches. Notre principale contribution dans ce projet de fin d'études de master à été la proposition d'une nouvelle version d'un protocole qui permet de garder l'aspect confidentiel des contributions, mais en même temps permet la rétractation des contributeurs dans le cas où ils le désirent.

En conclusion, cette exploration approfondie des DAO a enrichi notre compréhension de leur structure, de leur fonctionnement ainsi que des défis auxquels ils font face, notamment en matière de confidentialité. Les DAO sont en plein essor, et sont devenus des acteurs

incontournables dans la gouvernance et les organisation décentralisées. Leur étude est cruciale pour appréhender pleinement les répercussions de la technologie blockchain dans divers secteurs.

Bibliographie

[1] D. Yaga, P. Mell, N. Roby, and K. Scarfone, “Blockchain technology overview,” National Institute of Standards and Technology, Gaithersburg, MD, NIST IR 8202, Oct. 2018. doi: 10.6028/NIST.IR.8202.

[2] A. Narayanan, J. Bonneau, E. Felten, A. Miller, and S. Goldfeder, “Bitcoin and Cryptocurrency Technologies”.

[3] A. Joshi, M. Han, and Y. Wang, “A survey on security and privacy issues of blockchain technology,” *Math. Found. Comput.*, vol. 1, pp. 121–147, Jan. 2018, doi: 10.3934/mfc.2018007.

[4] B. Sriman, s Kumar, and S. Prabakaran, “Blockchain Technology: Consensus Protocol Proof of Work and Proof of Stake,” 2020, pp. 395–406. doi: 10.1007/978-981-15-5566-4_34.

[5] C. T. Nguyen, D. T. Hoang, D. N. Nguyen, D. Niyato, H. T. Nguyen, and E. Dutkiewicz, “Proof-of-Stake Consensus Mechanisms for Future Blockchain Networks: Fundamentals, Applications and Opportunities,” *IEEE Access*, vol. 7, pp. 85727–85745, 2019, doi: 10.1109/ACCESS.2019.2925010.

[6] Q. Hu, B. Yan, Y. Han, and J. Yu, “An Improved Delegated Proof of Stake Consensus Algorithm,” *Procedia Comput. Sci.*, vol. 187, pp. 341–346, Jan. 2021, doi: 10.1016/j.procs.2021.04.109.

[7] A. O. Bada, A. Damianou, C. M. Angelopoulos, and V. Katos, “Towards a Green Blockchain: A Review of Consensus Mechanisms and their Energy Consumption,” in *2021 17th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, Pafos, Cyprus: IEEE, Jul. 2021, pp. 503–511. doi: 10.1109/DCOSS52077.2021.00083.

[8] M. Fiore and M. Devesas Campos, “The Algebra of Directed Acyclic Graphs,” in *Computation, Logic, Games, and Quantum Foundations. The Many Facets of Samson Abramsky*, vol. 7860, B. Coecke, L. Ong, and P. Panangaden, Eds., in Lecture Notes in Computer Science, vol. 7860. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 37–51. doi: 10.1007/978-3-642-38164-5_4.

[9] M. Castro and B. Liskov, “Practical Byzantine Fault Tolerance”.

[10] A. Zabat and M. Mahamdioua, “Combinaison de blockchain et biométrie pour la gestion des identités,” Thesis, University of Jijel, 2020. Accessed: May 09, 2024. [Online]. Available: http://dspace.univ-jijel.dz:8080/xmlui/handle/123456789/8573

[11] H. T. M. Gamage, H. D. Weerasinghe, and N. G. J. Dias, “A Survey on Blockchain Technology Concepts, Applications, and Issues,” *SN Comput. Sci.*, vol. 1, no. 2, p. 114, Mar. 2020, doi: 10.1007/s42979-020-00123-0.

[12] A. A. Monrat, O. Schelen, and K. Andersson, “A Survey of Blockchain From the Perspectives of Applications, Challenges, and Opportunities,” *IEEE Access*, vol. 7, pp. 117134–117151, 2019, doi: 10.1109/ACCESS.2019.2936094.

[13] W. Baiod, J. Light, and A. Mahanti, “Blockchain Technology and its Applications Across

Multiple Domains: A Survey," *J. Int. Technol. Inf. Manag.*, vol. 29, no. 4, pp. 78–119, Jan. 2021, doi: 10.58729/1941-6679.1482.
[14] S. Wang, W. Ding, J. Li, Y. Yuan, L. Ouyang, and F.-Y. Wang, "Decentralized Autonomous Organizations: Concept, Model, and Applications," *IEEE Trans. Comput. Soc. Syst.*, vol. 6, no. 5, pp. 870–878, Oct. 2019, doi: 10.1109/TCSS.2019.2938190.
[15] T. Nielsen, "Cryptocorporations: A Proposal for Legitimizing Decentralized Autonomous Organizations," *UTAH LAW Rev.*, no. 5.
[16] S. Riva, "Decentralized Autonomous Organizations (DAOs) as subjects of law - The recognition of DAOs in the Swiss legal order," 2019. doi: 10.13140/RG.2.2.18198.22087.
[17] Y. Faqir, J. Arroyo, and S. Hassan, *An overview of decentralized autonomous organizations on the blockchain*. 2020, p. 8. doi: 10.1145/3412569.3412579.
[18] Galia Kondova and Renato Barba, "Governance of Decentralized Autonomous Organizations," *J. Mod. Account. Audit.*, vol. 15, no. 8, Aug. 2019, doi: 10.17265/1548-6583/2019.08.003.
[19] "How to Build a Private DAO on Ethereum - HackMD." Accessed: Apr. 26, 2024. [Online]. Available: https://hackmd.io/nCASdhqVQNWwMhpTmKpnKQ
[20] L. Liu, S. Zhou, H. Huang, and Z. Zheng, "From Technology to Society: An Overview of Blockchain-Based DAO," *IEEE Open J. Comput. Soc.*, vol. 2, pp. 204–215, 2021, doi: 10.1109/OJCS.2021.3072661.
[21] B. Schneider, R. Ballesteros, P. Moriggl, and P. M. Asprion, "Decentralized Autonomous Organizations – Evolution, Challenges, and Opportunities".
[22] Q. DuPont, M. Gkikaki, and C. Rowan, "DAO, Blockchain and Cryptography: A conversation with Quinn DuPont," 2020.
[23] A. Wright, "THE RISE OF DECENTRALIZED AUTONOMOUS ORGANIZATIONS: OPPORTUNITIES AND CHALLENGES".
[24] J. Bonneau, A. Miler, J. Clark, A. Narayanan, J. A. Kroll, and E. W. Felten, "Research Perspectives and Challenges for Bitcoin and Cryptocurrencies." 2015. Accessed: May 09, 2024. [Online]. Available: https://eprint.iacr.org/2015/261
[25] R. Zhang, R. Xue, and L. Liu, "Security and Privacy on Blockchain," *ACM Comput. Surv.*, vol. 52, no. 3, pp. 1–34, May 2020, doi: 10.1145/3316481.
[26] "Foundations of Cryptography - Volume 1 [Goldreich]." Accessed: Apr. 28, 2024. [Online]. Available: https://www.wisdom.weizmann.ac.il/~oded/foc-vol1.html
[27] "EIP-197: Precompiled contracts for optimal ate pairing check on the elliptic curve alt_bn128," Ethereum Improvement Proposals. Accessed: May 07, 2024. [Online]. Available: https://eips.ethereum.org/EIPS/eip-197
[28] D. Boneh, "The Decision Diffie-Hellman problem," in *Algorithmic Number Theory*, vol. 1423, J. P. Buhler, Ed., in Lecture Notes in Computer Science, vol. 1423. , Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 48–63. doi: 10.1007/BFb0054851.
[29] "Ganache | Overview - Truffle Suite." Accessed: May 12, 2024. [Online]. Available: https://archive.trufflesuite.com/docs/ganache/
[30] C. de Balasy, "Approfondir – Ethereum : smart contracts et Oracles," Coinhouse. Accessed: May 12, 2024. [Online]. Available: https://www.coinhouse.com/fr/blog/actualites/approfondir-ethereum-smart-contracts-et-oracles/
[31] F. Guenet, "Comment utiliser Remix Ethereum pour développer des Smart-Contracts," Tokize.com. Accessed: May 12, 2024. [Online]. Available: https://www.tokize.com/fr/remix-ethereum-smart-contracts/
[32] "Qu'est-ce que le JavaScript ? - Apprendre le développement web | MDN." Accessed:

May 12, 2024. [Online]. Available: https://developer.mozilla.org/fr/docs/Learn/JavaScript/First_steps/What_is_JavaScript
[33] "Tutoriel : intro à React – React." Accessed: May 12, 2024. [Online]. Available: https://fr.legacy.reactjs.org/tutorial/tutorial.html
[34] "Commencer avec MetaMask | MetaMask Help Center □♥." Accessed: May 21, 2024. [Online]. Available: https://support.metamask.io/fr/getting-started/getting-started-with-metamask/
[35] A. Rathod, "What is Web3.js: Basic to Advanced Guide for Developers," MXICoders INC. Accessed: May 21, 2024. [Online]. Available: https://www.mxicoders.com/web3js/
[36] "MySQL," *Wikipédia*. May 16, 2024. Accessed: May 21, 2024. [Online]. Available: https://fr.wikipedia.org/w/index.php?title=MySQL&oldid=215152223
[37] "XAMPP," *Wikipédia*. May 08, 2024. Accessed: May 21, 2024. [Online]. Available: https://fr.wikipedia.org/w/index.php?title=XAMPP&oldid=214925864
[38] "Express.js," *Wikipedia*. Apr. 12, 2024. Accessed: May 21, 2024. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Express.js&oldid=1218558568#Popularity
[39] "Définition Visual Studio Code - Bility - Agence de développement web sur-mesure." Accessed: May 21, 2024. [Online]. Available: https://bility.fr/definition-visual-studio-code/