République algérienne démocratique et populaire
الجـمهورية الجـــزائرية الـــديمقراطية الشـــــعبية

Ministre de l'enseignement supérieur et de la recherche scientifique
وزارة التـــعليم العـــالي والبـــــحث العـــــلمي

Université MOHAMED EL BACHIR EL IBRAHIMI - Bordj Bou Arreridj
جـــامعة مـــحمد البـــشير الإبـــراهيمي – بـرج بـوعـريـريـج

Faculté des Mathématiques et d'Informatique

Département d'informatique

MEMOIRE

Pour obtenir le diplôme de

Master 2 Réseaux et MultiMedia

Thème :

Développement d'une plateforme de e-commerce basée sur la technologie blockchain

Par

SALAKDJI Mostefa

Soutenu le : 14/07/2021 devant le jury composé :

Président	: M. SABRI Lyazid	M.C.A. à l'université de B.B.A.
Examinateur	: M. BENAOUDA Nadjib	M.C.B. à l'université de B.B.A.
Examinateur	: M. SENOUCI Oussama	M.C.B. à l'université de B.B.A.
Encadreur	: M. BEGHOURA Mohamed Amine	M.C.A. à l'université de B.B.A.

Promotion : 2020/2021

REMERCIMENTS

Louange A Dieu, le miséricordieux, sans lui rien de tout cela n'aurait pu être.

Je tiens à remercier vivement Mr BEGHOURA Mohamed Amine, pour m'avoir honoré par son encadrement, pour sa disponibilité, ses précieux conseils et ses encouragements qui ma permit de mener à bien ce travail. Je tiens à exprimer ma gratitude aux membres de jury pour avoir accepté de juger ce travail. Je remercie chaleureusement tous nos enseignants pour leurs conseils, leurs gentillesses, et leurs générosités. Un merci particulier à mes parents, pour leurs amours, leurs sacrifices et leurs patiences. Un énorme merci à ma familles et amis pour leurs éternel soutient et la confiance qu'ils ont en nos capacité.

DEDICACES

Je dédie ce modeste travail à ma famille et particulièrement à mes Parents pour leurs soutiens qu'ils m'ont accordés tout au long mon chemin.

A mes amis ainsi que toutes ces belles personnes que j'ai eu l'immense bonheur de connaitre au cours de ces années à l'université.

Une dédicace spéciale à Safia, Lotfi et ces enfants, la petite princesse Aya, Ahmed, Lokman, Yasser et toutes mes sœurs.

SALAKDJI Mostefa

Table des matières

Table de figures

ملخص

الأسواق المادية الحديثة هي أماكن للثقة والأمن حيث تُبنى الثقة على سمعة الأعمال السابقة، ولكن عندما يتعلق الأمر بالمجال الرقمي، لدينا سوق تقليدي مركزي معماريًا بدون شفافية أو مساءلة لحل مشاكل النمو المتزايد. من الصعب تطبيق نفس المبادئ مثل أسواق العالم الحقيقي بسبب النماذج المنهجية للتداول التي عفا عليها الزمن والتي لا تزال قيد الاستخدام. تستند المشكلة إلى إجماع هذا النظام المعقد حيث نحتاج إلى نظام بيئي منفصل لتوفير سجل مشارك للحكم على السمعة، ووسيط للإشراف على المعاملات وحل جميع النزاعات المعلقة وإعادة المعلومات من المعاملات الشفافة والموثوقة المتاحة للجميع. يمكن التخفيف من جميع المشكلات الرئيسية المذكورة أعلاه من خلال تطبيقBlockchain ، وهي بنية بيانات جديدة تستخدم الموزعة، أي تقنية دفتر الأستاذ اللامركزي للمساعدة في تأمين شبكات المعاملات.

يهدف FabMarket في هذه الرسالة إلى أن يكون تطبيق سوق تجارة إلكترونية موزع عبر الإنترنت لتبادل الأصول والبيانات والخدمات في شبكة عملاء من نظير إلى نظير، مدعومة بعقود ذكية مخصصة ومصممة لاستخدامHyperledger Fabric

Abstract

Modern physical markets are places of trust and security where trust is built on the reputation of previous acts, but when it comes to the digital realm, we have a traditional market that is architecturally centralized with no transparency or accountability to solve the problems of a growing audience and having an it is difficult to apply the same principles as real world markets due to outdated trading methodological models that are still in use. The problem is based on theconsensus of this complex system where we need a separate ecosystem to provide a participanthistory to judge reputation, an intermediary to oversee transactions and resolve all pending disputes and return information from transparent and reliable transactions available to all. All of the above key issues can be mitigated by applying Blockchain, a new data structure that uses distributed, i.e. decentralized ledger technology to help secure transaction networks. FabMarket in this dissertation aims to be a distributed online e-commerce marketplace application for the exchange of assets, data, and services in a peer-to-peer customer network, which is powered bycustom smart contracts and built to using a Hyperledger Fabric.

Résumé

Les marchés physiques modernes sont des lieux de confiance et de sécurité où la confiance est fondée sur la réputation d'actes antérieurs, mais en ce qui concerne le domaine numérique, nous avons un marché traditionnel centralisé sur le plan architectural sans transparence ni responsabilité pour résoudre les problèmes d'un public croissant et avoir un il est difficile d'appliquer les mêmes principes que les marchés du monde réel en raison de modèles de méthodologies commerciales obsolètes qui sont toujours utilisés. Le problème est basé sur le consensus de ce système complexe où nous avons besoin d'un écosystème séparé pour fournir un historique des participants pour juger de la réputation, un intermédiaire pour superviser les transactions et résoudre tous les litiges en cours et rendre les informations de transaction transparentes et fiables à la disposition de tous. Tous les problèmes clés ci-dessus peuvent être atténués en appliquant Blockchain, une nouvelle structure de données qui utilise la technologie des registres distribués, c'est-à-dire décentralisés, pour aider à sécuriser les réseaux de transactions. FabMarket dans ce mémoire vise à être une application de marché de commerce électronique en ligne distribuée pour l'échange d'actifs, de données et de services dans un réseau de clients pair à pair, qui est alimenté par des contrats intelligents personnalisé et construit à l'aide d'un Hyperledger Fabric.

Introduction Générale

Introduction Générale

Un système blockchain est un grand livre distribué où toutes les transactions qui ont eulieu dans le réseau sont stockés dans des blocs. Ces blocs sont triés et ajoutés au réseau un parun, formant une chaîne de blocs.

Le système est décentralisé car le Ledger est répliqué au sein des nœuds participant autraitement prestations de service. Les propriétés de sécurité de ce système décentralisé sont basées sur des techniques cryptographiques qui rendre cette chaîne de blocs immuable. La génération de nouveaux blocs et le processus de réplication du Ledger au sein le nœud du réseau est régi par l'algorithme de consensus défini dans le réseau. Le consensus L'algorithmeutilise des fonctions cryptographiques et des règles métier pour décider quel nœud ajoutera leprochain bloc et la réplication du dernier état de la blockchain sur le reste des nœuds.

Dans tout système, il est crucial de connaître l'état de l'infrastructure et son utilisation (exploitation, utilisateurs, requêtes, entre autres) pour mesurer les performances du service quis'exécute dessus. Dans le cas de système décentralisé, cela peut être plus délicat que dans un système central car le statut dépendra du nœud et il y a plusieurs caractéristiques qui l'ont rendu plus complexe. Dans le cas de la blockchain, la mesure le statut différent du Ledger dépendra des multiples composants qui donnent vie au décentralisé système.

Une simple requête pour obtenir des informations d'un bloc que nous devons connecterà un canal spécifique (Ledger) et trouvez le bloc. Dans le cas des transactions, ces opérations peuvent être plus complexes car elles dépendent du statut. Si nous souhaitons connaître le statut d'une transaction simulée, une transaction endossée ou la transaction engagée, il faudraitprendre en compte différents modules d'Hyperledger Fabric, comme : consensus (responsable du processus de confirmation transactionnelle et de génération de blocs), état de la base de données, statut et service d'adhésion, afin de savoir qui est valablement autorisé à effectuer des instructions.

Hyperledger Fabric fournit un cadre avec des instructions standard pour fonctionnersur le réseau appelé Code de chaîne. C'est l'un des éléments clés de la mise en œuvre de la blockchain, car il nous donne une interface pour communiquer avec différents nœuds et
canaux dans le système décentralisé. De plus, nous avons Hyperledger Fabric qui est une suitepour gérer et développer de nouvelles applications pour la blockchain de manière conviviale environnement utilisateur.

Dans les sections suivantes, nous décrirons les concepts introduits ci-dessus et comment tous les composants peuvent interagir en utilisant Chaincode et Hyperledger Fabric,et pour cela on créer un Web Site E-commerce comme une source d'informations à enregistrer dans le Ledger.

Chapitre 1

Vue générale sur le E-Commerce

1. Introduction

Ce chapitre est consacré à la définition du e-commerce, à examiner son historique,ainsi qu'à identifier ses différentes formes.

Comprendre cette nouvelle pratique nous ramène inévitablement à préciser ses particularités par rapport au commerce traditionnel et à identifier les différents intervenants dans une transaction électronique.

2. Définition du e-commerce

Le e-commerce ou le commerce électronique (un sous ensemble de l'e-business) utilise des moyens électroniques pour réaliser l'achat, la vente et l'échange de biens et de services sur des réseaux informatiques comme internet. Pour cette raison, le E-Commerce dématérialise l'ensemble de procédures liées aux activités commerciales comme le payement en ligne qui est considéré comme un bon exemple de la dématérialisation de la monnaie au profit de transactions informatiques/électroniques/numériques. (1)

3. E-commerce && commerce traditionnel

3.1. Commerce électronique par rapport au commerce traditionnel

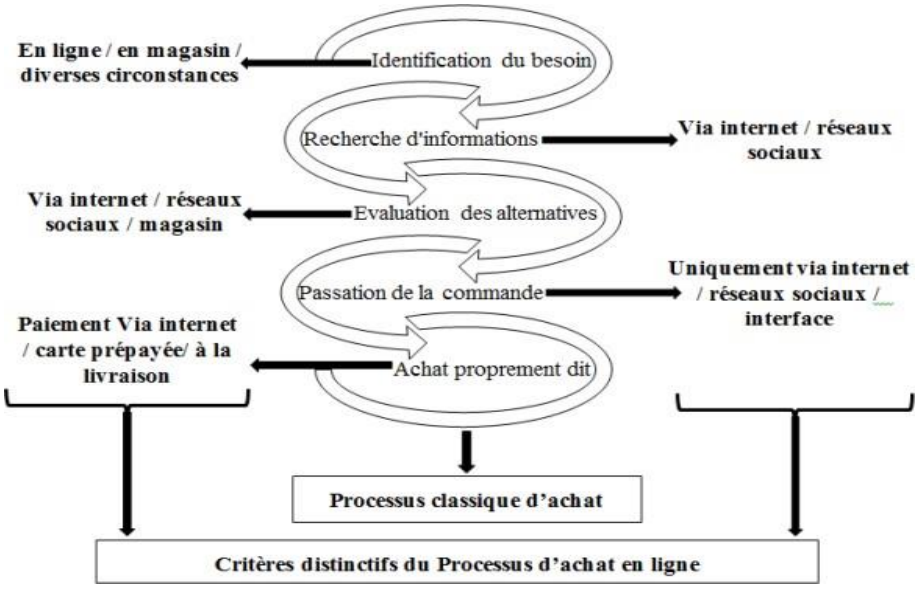


Figure 1 : processus commerce traditionnel et le e-commerce (2)

Les transactions en ligne offrent de nombreux avantages que le commerce traditionnel ne permet pas, notamment la rapidité, la réduction importante du cycle de vente et la réduction des coûts.

3.2. Les différents intervenants dans une transaction électronique [3]

Une vision globale du e-commerce nous donne trois types d'acteurs :

• Les clients : sont ceux qui désirent acquérir un bien ou un service pour satisfaire un besoin quelconque. Ils peuvent être particuliers, ou entreprises ; d'envergure nationale ou internationale.
Ces clients peuvent effectuent des achats en ligne en utilisant des cartes de crédits.

• Les vendeurs : ce sont ceux qui, possédant un bien ou ayant la capacité de produire un service, sont désireux de se départir de ce bien ou de fournir ce service moyennant une rémunération et utilisant, à cette fin des supports informatiques et électroniques.

Les vendeurs gèrent la commercialisation de leurs produits à travers des sites web (sites marchands).

• Les intermédiaires : ce sont tous ceux qui, par l'intermédiaire des supports informatiques, facilitent ou établissent le processus de transaction commerciale entre clients et vendeurs. Il s'agit principalement :

- Des intermédiaires techniques : fournisseurs d'accès Internet, responsables de la publication des informations des vendeurs, la disponibilité des sites et l'honnêteté des informations transmises par les clients.

- Des intermédiaires financiers : les émetteurs de cartes de crédits qui effectuent les transferts d'argent du compte du client à celui de l'entreprise (vendeur).

3.3. Le déroulement d'une transaction électronique [3]

Une transaction électronique est l'autorisation donnée par le porteur d'une carte de paiement électronique d'effectuer un certain type d'opération au bénéfice du marchand, depuisle compte associé à sa carte bancaire et géré par son institution financière.

Le schéma suivant détaille les différentes étapes d'une transaction électronique :

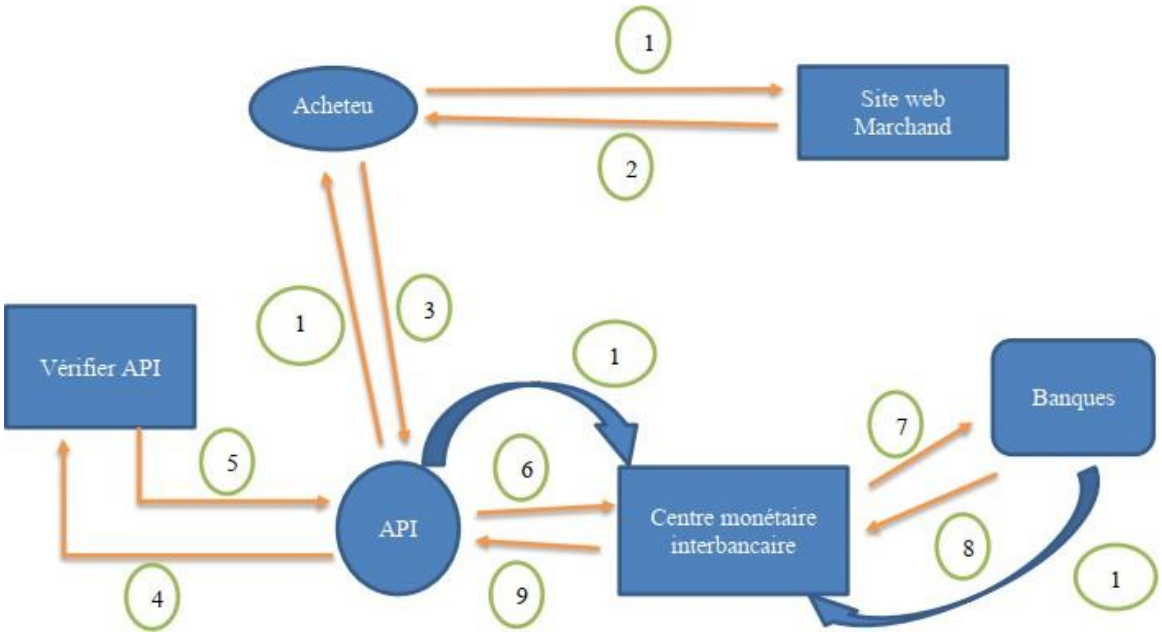


Figure 2 : les différentes étapes d'une transaction

électroniqueEtape 1 : Achat de biens ou de services

Le client se connecte à un site marchand et procède à la sélection des articles à acheterou aux créances à régler.

Etape 2 : Confirmation de la commande

Une fois son choix validé, il confirme son attention de payer par carte bancaire en cliquant sur le bouton « Payer » et il sera orienté automatiquement vers la page de paiement sécurisée de l'intermédiaire.

Etape 3 : Saisie des données de paiement

L'intermédiaire vers sont API reçoit et vérifie la conformité de la demande de paiement reçue du site marchand et affiche au client un écran de paiement personnalisé.

Etape 4 et 5 : Demande d'authentification de la carte et renvoie la réponse de l'émetteur à lademande d'authentification.

Etape 6 à 9 : Demande d'autorisation et réponse

En temps réel, une demande d'autorisation est envoyée par l'API qui la transmet via son réseau Interbancaire à la banque du porteur de la carte. Cette dernière accepte ou refuse la demande d'autorisation et retourne le résultat de l'autorisation.

Etape 10 : Répercussion de la réponse sur le client

L'API vérifie, enregistre le résultat de l'autorisation et affiche en temps réel une réponse au client :

- *Reçu du paiement si réponse positive, c'est-à-dire si l'autorisation a été accordée et acceptée.*
- *Message de refus le cas échéant. Le client sera invité à refaire sa demande de paiement.*

Etape 11 : Confirmation de la transaction au

CMI.Etape 12 : Règlement de la transaction

Une fois la transaction validée par le commerçant le CMI procède au règlement en débitant le client et en créditant le commerçant, Bref, on peut récapituler le déroulement d'unetransaction électronique en 3 phases :

• Shopping : le client et le marchand se mettent d'accord à travers un site marchand surun/ensemble de biens à acheter et sur le montant à payer par le client.

• Paiement : l'intermédiaire financier procède au règlement de la transaction après l'authentification de la carte de crédit et l'obtention d'une autorisation de paiement auprès dela banque du client.

• Livraison : au terme de la transaction de paiement le marchand rend au client les biens/services préalablement sélectionnés.

4. Les types du e-commerce [(1)]

Les applications de E-commerce peuvent être réparties en quatre catégories principales : B2B, B2C, C2B et C2C.

4.1. B2B (Business-to-Business)

Ceux sont les entreprises qui font affaire avec d'autres, comme les fabricants qui vendent à des distributeurs et grossistes, qui a leur tour vendent aux détaillants. La tarification est basée sur la quantité de l'ordre et est souvent négociable.

4.2. B2C (Business-to-Consumer)

Ceux sont les entreprises vendant au grand public en général grâce a des catalogues en utilisant des logiciels panier. En volume en dollars, B2B à la palme, cependant B2C est vraiment ce que l'utilisateur, a en tête en ce qui concerne le commerce électronique, dans son ensemble.

4.3. C2B (Consumer-to-Business)

Le consumer to business (C2B) est un modèle d'entreprise (business model) dans lequel les consommateurs (les particuliers) sont au service de l'entreprise en apportant un produit ou une prestation, et non le contraire comme c'est le cas traditionnellement.

4.4. C2C (Consumer-to-Consumer)

Il existe de nombreux sites offrants de petites annonces gratuites, enchères, et des forums ou les particuliers peuvent acheter et vendre en ligne grâce au système de paiements tels que PayPal [2], ou les gens peuvent envoyer et recevoir de l'argent en ligne en toute simplicité. Le service d'enchère d'eBay est un bon exemple de commerce de personne, des transactions ont lieu tous les jours depuis 1995.

5. Technologies informatiques & E-commerce [3]

Plusieurs technologies sont utilisées dans le e-commerce pour permettre l'interopérabilité informatique, entre les systèmes informatiques des clients et des fournisseurs, ainsi que des établissements financiers qui interviennent dans les paiements.

Ce type d'interopérabilité repose généralement sur l'emploi de métadonnées pour le partage des données et sur l'utilisation des hyper cubes d'analyse qui permettent une synchronisation en temps réel des différents systèmes.

En outre, le développement du e-commerce en Algérie doit passer par l'amélioration de la qualité de l'accès au réseau de télécommunications en augmentant la largeur de la bande passante et en garantissant la disponibilité de la connexion à internet.

Les domaines qui peuvent être considérés comme des technologies pouvant modifier les possibilités du e-commerce sont : le Web sémantique, le Web pervasif et le marketing nomadeLe Web sémantique, qui complète les balises HTML par des balises porteuses de sens permet de trouver très simplement et très rapidement ce que l'on cherche parmi la colossale quantité de données qui sera produite chaque année.

6. Les avantages du e-commerce [2]

Le e-commerce permet d'avoir une boutique ouverte 24h/24 tous les jours de l'année grâce à l'utilisation des sites web marchants. Ces sites de ventes en ligne possèdent des systèmes de paiements de plus en plus sécurisés et ne diffuse pas les informations personnelles de leurs clients. D'autre part, les sites « comparateurs de prix » permettent d'avoir le prix est plus attractif sur Internet. En outre, les vendeurs de produits et les fournisseurs de services qui utilisent le e-commerce peuvent proposer des tarifs très compétitifs car ils peuvent réduire la facture de leurs frais de fonctionnement tout en garantissant des délais de livraison de plus en plus très courts. Dans la section qui suit, nous allons analyser les avantages que le e-commerce procure à l'entreprise ainsi qu'à ses clients.

6.1. Pour les entreprises

- Il permet de couvrir des marchés dont l'atteinte était trop difficile par lesmoyens classiques de commercialisation.

- Il apporte une plus forte convivialité par rapport à la commande à distancetraditionnelle grâce aux formulaires interactifs (images, sons et animations).
- Il favorise le développement d'une relation personnelle avec le consommateuren envisageant des politiques de fidélisation du client.
- Il permet de réduire les prix publics des produits en éliminant la marge laisséehabituellement aux intermédiaires.
- L'enregistrement des données de ventes par le site de e-commerce est quasiment automatique et demande peu d'effort de la part des gestionnaires.

6.2. Pour les clients

- L'E-commerce est un excellent outil de présélection sur internet.
- Il permet la recherche du meilleur prix à l'aide des sites comparateur de prix.
- Il élimine la de pression exercée par les vendeurs pour la vente de leurs produits.
- Le e-commerce fournie un marché de proximité à l'échelle mondiale.
- Il offre un gain de temps considérable.
- Le e-commerce permet d'avoir une offre actualisée (mise à jour régulière).
- Il facilite la promotion de nouveaux produits au niveau des clients.
- Il offre la possibilité de passer des commandes spécifiques pour ses clients

7. Les inconvénients du e-commerce [(2)]

7.1. Pour les entreprises

- Les entreprises qui ont adopté ce mode rencontrent une résistance psychologique chez certains de ses clients.
- L'incertitude et le manque de confiance autour de la sécurisation des moyens de paiement bien que les méthodes de cryptage de données assurent une confidentialité quasi parfaite lors des transactions.
- La résistance des intermédiaires (grossistes, distributeurs) qui craignent une perte d'emplois et une diminution de leurs chiffres d'affaires.

7.2. Pour les clients

- Il permet le pistage informatique à partir des cookies qui peuvent retracer toutes leshabitudes du consommateur.
- L'insécurité des paiements et la peur de tomber sur un cybermarchand mal honnête quine livre pas.
- Le manque de relations humaines et le sentiment d'isolement devant sa machine
- Le manque de contact avec le produit.
- Les difficultés de recours en cas de problème.
- Peut engendrer une dépendance

8. Conclusion

Les cryptomonnaies deviennent de plus en plus utiles pour les achats quotidienssur les sites E-Commerce.

Nombreuses sont les crypto-monnaie qui utilise la technologie Blockchain (Bitcoin, Ethereum, Ripple, etc). Cette technologie offre une sécurité, rapidité des transactions et gains de productivité et d'efficacité.

Le chapitre prochain sera sur technologie Blockchain, utilisation, domaine d'application avantages et inconvénients et l'application de cette technologie sur notre projet.

Chapitre 2

Etat de l’art de la blockchain

1. Introduction

Dans le chapitre précédent on a vu le E-Commerce et sa domination par rapport au commerce traditionnel, suite à cette évolution, il est évident que les technologies d'implémentation s'adaptent aux besoins actuels,

Dans ce chapitre on va concentrer sur la partie sérialisation des données, cette partie est considérer très important dans le domaine, ce sont les informations personnelles des clients oudes entreprises en jeu, ce concept-là a poussé les développeurs ou les chercheurs a trouvé un moyen d'enregistrer les données d'une façon sécurisé, anonyme et infalsifiable et la technologieBlockchain réponds à ces besoins.

2. Comment choisir la technologie Blockchain ?

La vraie question à se poser est de savoir avant toute chose si vous ont a réellement besoin d'une blockchain pour réaliser le cas d'usage. Pour cela, l'arbre de décision ci-dessous nous donnera toutes les réponses aux questions.

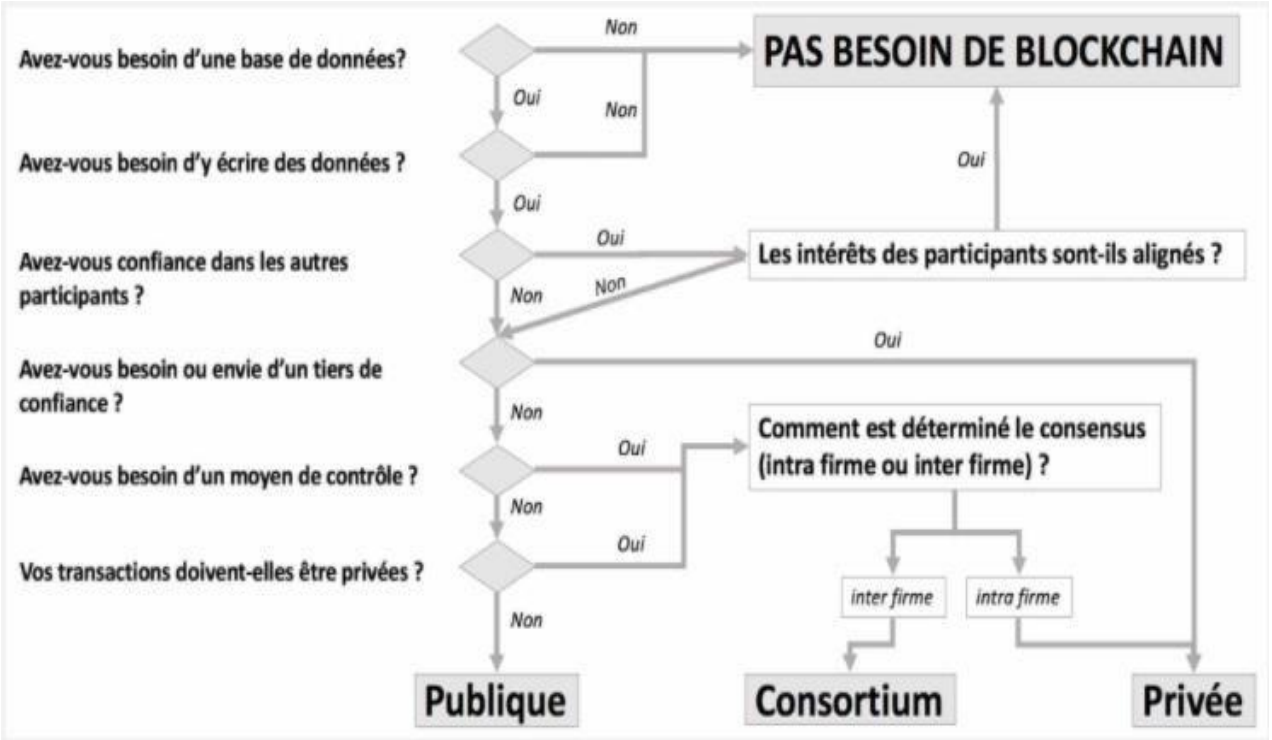


Figure 1 : Arbre de décision d'utilisation de la technologie Blockchain [(4)].

3. La Blockchain

La blockchain, ou chaîne de blocs, désigne originellement la structure de données utilisée par Bitcoin pour lister l'ensemble des transactions réalisées par ses utilisateurs depuis son commencement : les transactions sont regroupées dans des blocs chaînés entre eux, et des blocs sont rajoutés à la chaîne au cours du temps. Ce registre est partagé entre les membres d'unréseau, d'où le fait qu'on parle parfois de registre distribué.

Depuis 2015, la blockchain désigne également la technologie de consensus décentralisé remise au goût du jour par Bitcoin. Cette technologie désigne donc l'ensemble des méthodes permettant aux participants d'un réseau distribué de se mettre d'accord sans recourir à un tiers de confiance. (6)

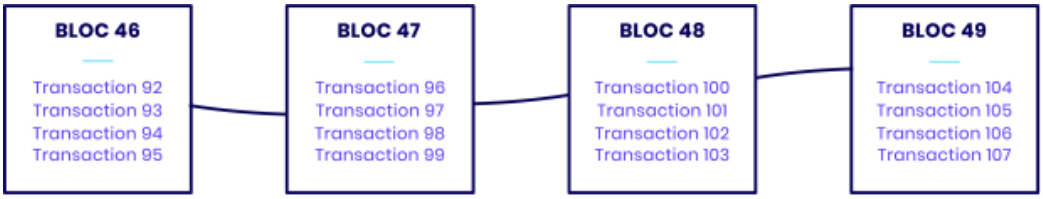


Figure 2 : chaîne de blocs (5).

4. Origine de la blockchain

Le concept de blockchain tel qu'on le connaît aujourd'hui a été inventé par Satoshi Nakamoto et décrit au sein du livre blanc de Bitcoin le 31 octobre 2008. Satoshi Nakamoto n'a néanmoins pas forgé le terme et on doit cela à Hal Finney, qui l'a utilisé pour la première fois dans sa réponse à Satoshi du 7 novembre 2008 pour désigner la chaîne de preuve de travail de Bitcoin. Il l'orthographiait alors « block chaine », en deux mots. Le terme a ensuite été repris par Satoshi dans le code source de la version 0.1 de Bitcoin, et il s'est progressivementpopularisé au sein de la communauté.

Cependant, la technologie derrière la blockchain n'a pas été entièrement créée avec Bitcoin et les éléments qui la composent sont bien plus anciens qu'on ne le croit : la blockchainest en effet le fruit de nombreuses années de recherche en économie, en informatique eten cryptographie.

La blockchain promeut l'idée de la décentralisation, qui, sans surprise, s'oppose à la centralisation. Aujourd'hui, nos échanges, transactions actuellement sont centralisées. Elles sont régies par les États, les banques, les grandes entreprises...

Ces acteurs auxquels nous nous référons sont ce qu'on appelle des tiers de confiance. Lablockchain, elle, fonctionne sans autorité centrale.

La blockchain permet de se passer de tiers de confiance pour la transmission de valeur entre deux entités. Le Bitcoin est la première application concrète de cette technologie. Il est maintenant possible, grâce à Bitcoin et à la blockchain, de transférer de la valeur sur internet entre deux entités sans intermédiaire. (6)

5. Exemple concret de l'application Blockchain (7)

On veut envoyez un fichier à un ami sur internet, ce dernier est dupliqué. On possède une copie du fichier et notre ami aussi : le fichier n'est pas unique. Le fichier ne part pas d'un point A pour arriver à un point B. Il reste au point A et une copie apparaît au point B. Cela ne peut donc pas fonctionner pour de la monnaie. Quand on donne 1 euro à un boulanger pour acheter une baguette, une copie de la pièce ne reste pas dans notre poche. La pièce n'est pas dupliquée, elle est unique, elle quitte notre porte-monnaie pour aller dans la caisse du boulanger.

Il était donc impossible d'utiliser les systèmes de transferts de valeur existants sur internet dans le cadre de la mise en place d'un système monétaire sur le web. La monnaie auraitperdu de la valeur, car à chaque échange elle aurait été dupliquée. Toute la difficulté était doncde réussir à créer une monnaie sur internet qui puisse fonctionner comme dans la vie de tous les jours.

Grâce à Bitcoin, cela est maintenant possible. Un système monétaire fonctionnel, fiableet sans tiers de confiance a vu le jour.

6. Déférents type de la blockchain :

La blockchain est une technologie de stockage et de transmission d'informations, transparente, sécurisée, et fonctionnant sans organe central de contrôle (définition de Blockchain France). Elle est la technologie au cœur du Web Décentralisé et de son corollaire, lafinance décentralisée.

Il existe des blockchains publiques, ouvertes à tous, et des blockchains privées, dont l'accès et l'utilisation sont limitées à un certain nombre d'acteurs.

Comme on a vue avant, il existe différents degrés de décentralisation, ce qui explique latrès grande diversité de registres distribués présente dans l'écosystème.
À l'origine, avec l'apparition de Bitcoin en janvier 2009, la blockchain est publique. Chaque utilisateur peut publiquement vérifier le registre des transactions, et chaque mineur disposant d'une puissance de calcul suffisante peut y ajouter des transactions en échange d'une récompense, ce qui assure la pérennité du système.

Néanmoins, avec le succès de Bitcoin et des premières cryptomonnaies, certains systèmes appelés « blockchains privées » ont commencer à émerger. Celles-ci affectent la transparence et l'absence de permission des blockchains publiques.

Le tableau suivant contient les permission écriture/lecture pour chaque type de blockchain (publique et privé) : [8]

	Écriture ouverte	**Écriture fermée**
Lecture ouverte	Blockchain publique	Blockchain à écriture permissionnée : les nœuds validateurs sont choisis préalablement (preuve d'autorité). Exemple : Ripple.
Lecture fermée	(Cas très rare) Blockchain à lecture permissionnée : les personnes pouvant lire le registre ont besoin d'une autorisation.	Blockchain à lecture et écriture permissionnée : les acteurs pouvant lire le registre et ceux pouvant valider les transactions sont présélectionnés. Exemples : IBM Food Trust, Tradelens.

Le schéma suivant montre l'architecture d'une blockchain publique et blockchain privé :

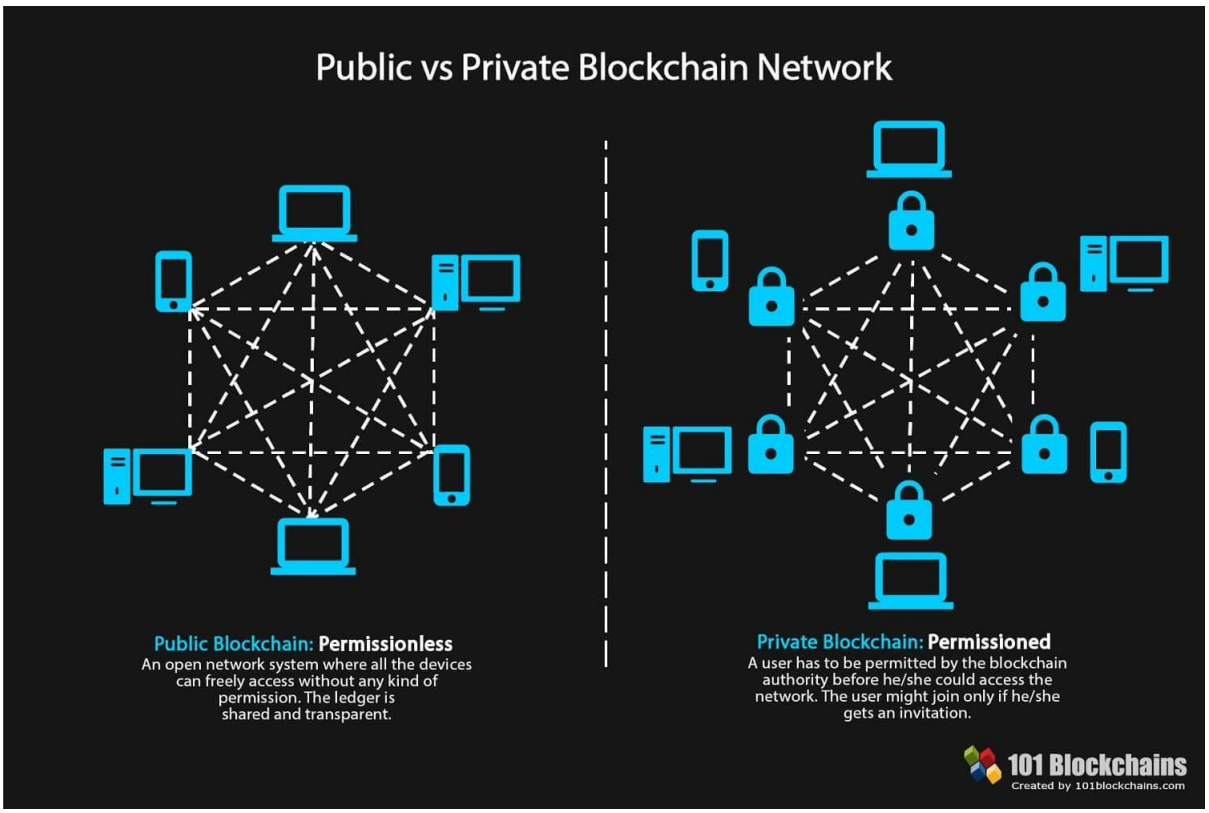


Figure 3 : Architecture blockchain privé et public.

6.1. Blockchain publique

Une blockchain publique est une chaîne qui soit à la fois visible et modifiable de manière publique par tous. N'importe qui peut observer le registre, notamment en téléchargeant le logiciel open-source correspondant. De plus, en devenant un nœud du réseau, la personne pourra vérifier l'intégralité des transactions historiques ainsi que les nouvelles transactions entrantes.

De même, n'importe qui peut devenir validateur à condition de dépenser de l'énergie (preuve de travail, minage) ou de posséder des jetons sous séquestre (preuve d'enjeu, forgeage).Cette ouverture permet au système résultant d'être robuste, c'est-à-dire de pouvoir continuer à fonctionner même si l'un des participants quitte le réseau. Tel que l'écrit Satoshi dans le livre blanc de Bitcoin, « les nœuds peuvent quitter et rejoindre le réseau quand bon leur semble. »

Les blockchains publiques font toutes intervenir une crypto-monnaie : il s'agit en effet de systèmes crypto-économiques qui reposent sur la valorisation d'un actif numérique pour pouvoir fonctionner.

Parmi les blockchains publiques on retrouve :

- Bitcoin : il s'agit de la première blockchain publique, que ce soit en termes de chronologie ou d'importance.
- Ethereum : c'est une plateforme de smart contracts dont les opérations sont stockées surune blockchain publique validée par
- Monero : blockchain publique particulière, puisqu'elle recense des transactions indéchiffrables, si bien que le monero (XMR) est considéré comme une cryptomonnaieanonyme.
- Tezos : à l'instar d'Ethereum, Tezos est une plateforme de contrats autonomes. Ce qui la distingue est qu'il s'agit d'une blockchain validée par preuve d'enjeu (liquide),contrairement aux trois blockchains précédemment citées.
- EOS : EOS est une blockchain dédiée aux smart contracts validée par preuve d'enjeu déléguée, c'est-à-dire que les validateurs sont sélectionnés en fonction du nombre de jetons EOS qu'ils possèdent et qui leur ont été délégués. Pour assurer des performances élevées, le système n'autorise que 21 producteurs de blocs, ce qui le place à la limite entre blockchain publique et blockchain privée.

Comme l'écrit le mathématicien Jean-Paul Delahaye, il faut s'imaginer « un très grand cahier, que tout le monde peut lire librement et gratuitement, sur lequel tout le monde peut écrire, mais qui est impossible à effacer et indestructible. »

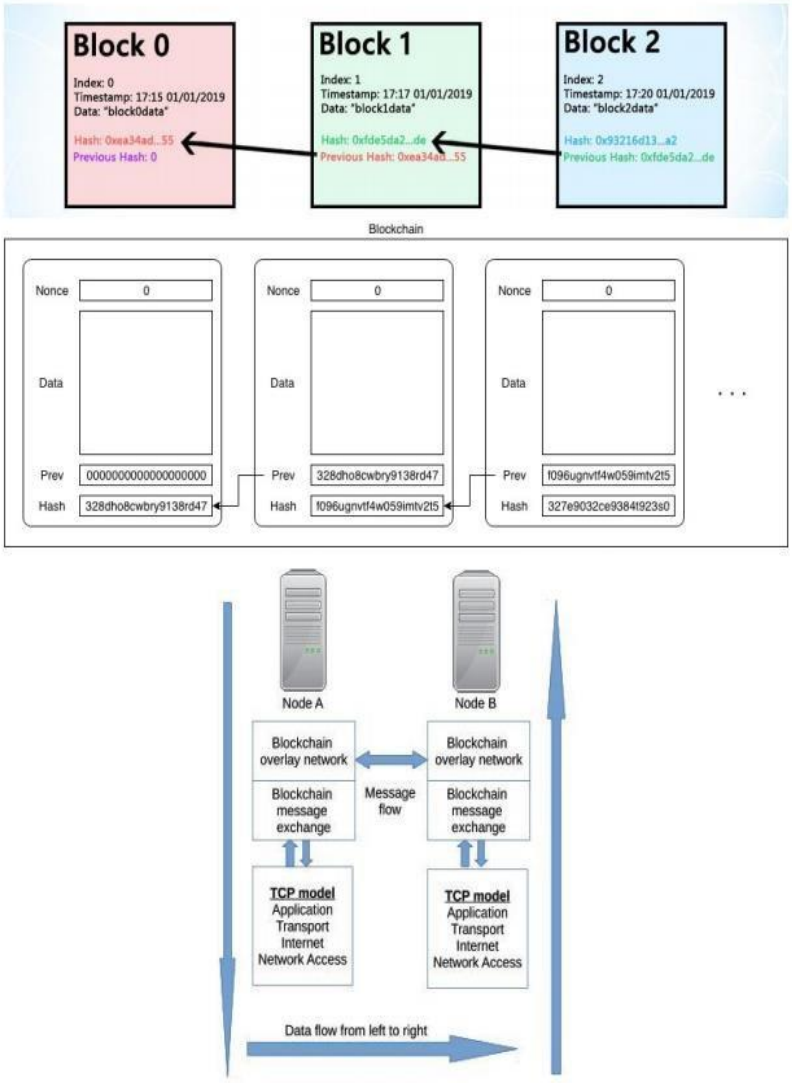


Figure 4 : Architecture de la Blockchain publique.

Source : blockchainhub.net

6.2. Blockchain privée

Une blockchain privée ou permissionnée est un registre distribué dont le contenu n'est pas disponible publiquement et / ou dont la validation est soumise à des permissions préétabliespar une autorité (preuve d'autorité). Pour mieux dire, la lecture et l'écriture de la chaîne peuventêtre restreintes : soit seule la lecture est restreinte, soit seule l'écriture est restreinte, soit les deux.

La blockchain privée est plus adaptée au monde de l'entreprise, qui est bien souvent effrayé par la transparence et par l'absence de permission. Ces systèmes ne nécessitent pas

nécessairement de jeton numérique pour fonctionner. La plupart du temps, leurs usages tournentautour de la traçabilité, de la chaîne logistique, de l'identité décentralisée.

Dans les modèles utilisés pour construire des blockchains privées, on retrouve : La suite d'outils Hyperledger, qui met à disposition des moyens de construire sa propre blockchain de manière rapide, dont notamment : Hyperledger Fabric et Hyperledger BESU.

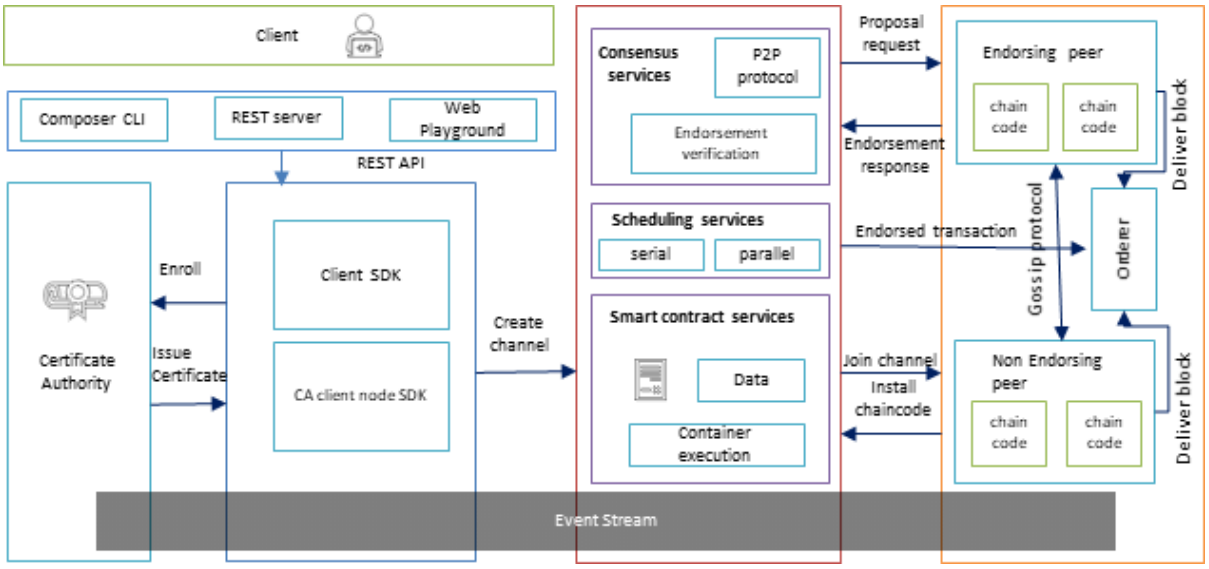


Figure 3 : Architecture blockchain privé [9].

7. Hyperledger :

Hyperledger est un projet mondial de blockchain d'entreprise qui offre le cadre, les normes, les directives et les outils nécessaires pour créer des blockchains open source et des applications associées à utiliser dans divers secteurs. Les projets d'Hyperledger incluent une variété de plates-formes de blockchain autorisées prêtes à l'emploi, où les participants au réseau se connaissent et ont donc un intérêt intrinsèque à participer au processus de consensus.

En utilisant les composants disponibles sous l'égide d'Hyperledger, une entreprise peut appliquer diverses solutions et services de blockchain modulaires pour améliorer considérablement les performances de ses opérations et l'efficacité de ses processus commerciaux.

Le projet Hyperledger a été créé en décembre 2015 par la Linux Foundation basée à San Francisco, Californie. Il a commencé avec 30 sociétés membres et compte aujourd'hui plus de120 sociétés membres [10].

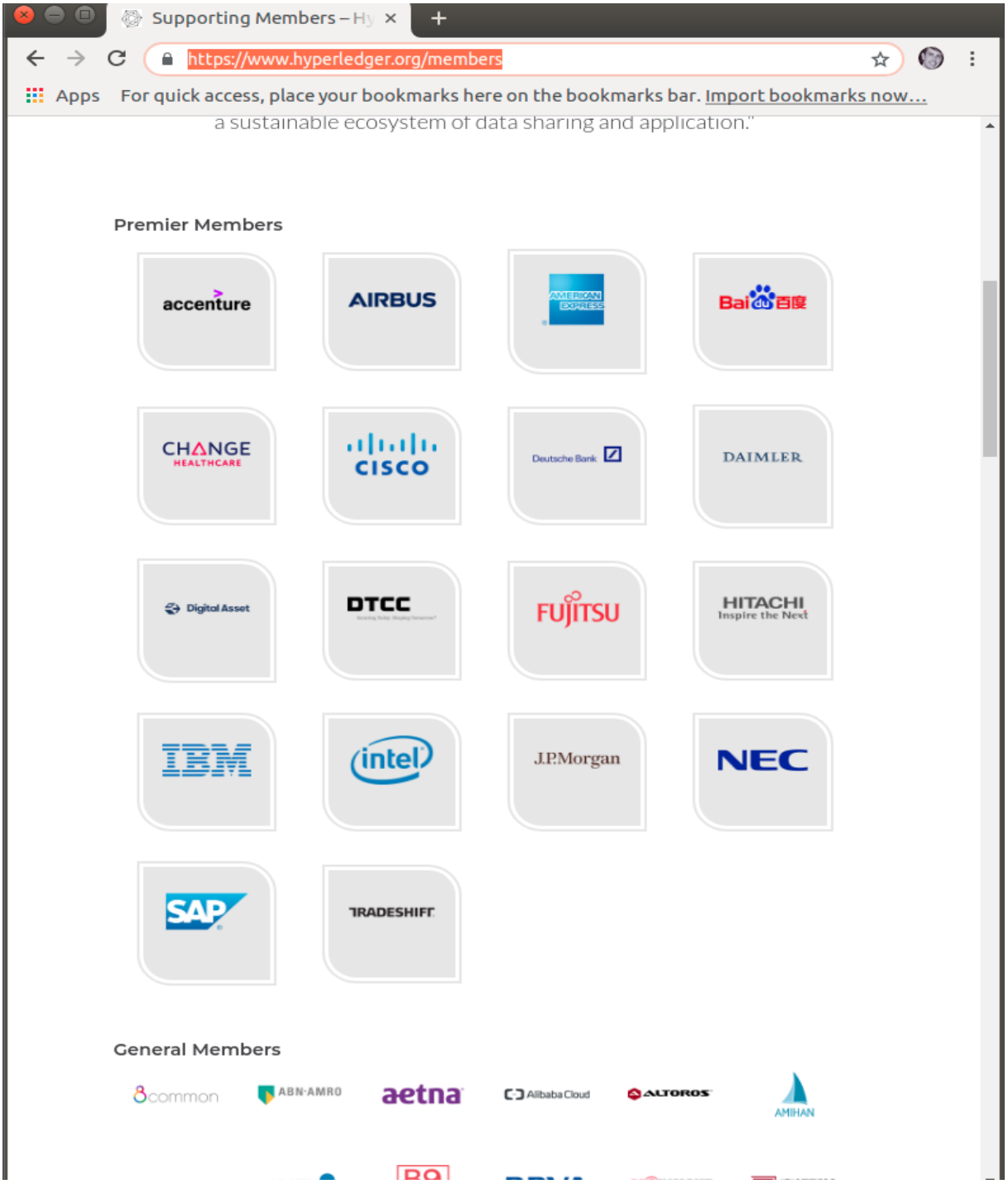


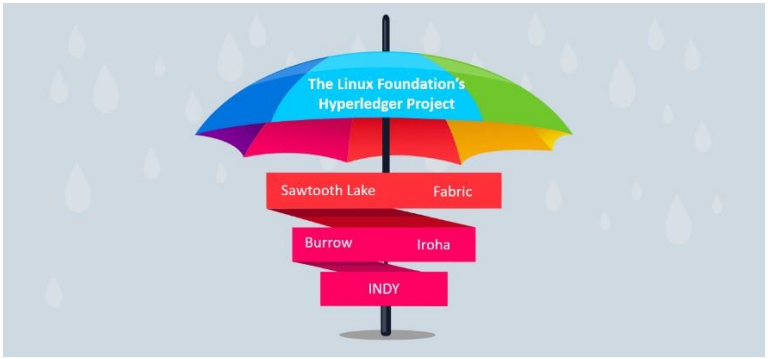
Figure 4 : Quelque sociétés membre dans le projet HyperLedger

7.1. Structure organisationnelle d'Hyperledger [11]

La Linux Foundation a fondé le projet Hyperledger en 2015 pour faire progresser les technologies de blockchain intersectorielles. Plutôt que de déclarer une norme blockchain unique, il encourage une approche collaborative du développement de technologies blockchainvia un processus communautaire, avec des droits de propriété intellectuelle qui encouragent ledéveloppement ouvert et l'adoption de normes clés au fil du temps.

Essentiellement, Hyperledger n'est pas une organisation, un réseau de crypto-monnaie ou un système de blockchain. Il ne prend pas en charge une crypto-monnaie comme le bitcoin, mais il fonctionne en fournissant l'infrastructure et les normes nécessaires au développement de divers systèmes et applications basés sur la blockchain à usage industriel. Considérez Hyperledger comme un hub, où divers projets et outils individuels basés sur la blockchain qui adhèrent à sa philosophie de conception définie fonctionnent sous son égide.

7.2. Déférents projets Hyperledger [11]



Les différents projets sont les suivants :

7.2.1. Hyperledger Fabric :

Le framework le plus populaire en ce moment et c'est ce que nous allons détailler dans le prochain chapitre.

Hyperledger Fabric est un cadre de blockchain modulaire qui sert de base au développement de produits, de solutions et d'applications basés sur la blockchain à l'aide de composants plug-and-play destinés à être utilisés dans des entreprises privées.

7.2.2. Hyperledger Explorer :

Est un utilitaire de tableau de bord qui permet la surveillance, la recherche et lamaintenance des développements de la blockchain et des données associées. 5

7.2.3. Hyperledger Burrow :

Est un nœud de blockchain de contrat intelligent Ethereum autorisé qui gère les transactions et exécute le code de contrat intelligent sur la machine virtuelle Ethereum (EVM).

7.2.4. Hyperledger Sawtooth :

Est une plate-forme de blockchain modulaire autorisée au niveau de l'entreprise quiutilise un algorithme de consensus innovant de preuve de temps écoulé . 7

7.2.5. Hyperledger Caliper :

Est un outil de référence de blockchain qui est utilisé pour évaluer les performancesd'une implémentation de blockchain spécifique.

7.3. Couches de technologie Hyperledger

En termes d'architecture, Hyperledger utilise les composants métier clés suivants :

- La couche consensus se charge de créer un accord sur la commande et de confirmerl'exactitude de l'ensemble des transactions qui constituent un bloc.
- La couche de contrat intelligent est chargée de traiter les demandes de transaction etd'autoriser uniquement les transactions valides.
- La couche de communication s'occupe du transport des messages d'égal à égal.
- Les services de gestion des identités sont la fonction nécessaire pour maintenir et validerles identités des utilisateurs et des systèmes et établir la confiance dans la blockchain.
- L'API, ou interface de programmation d'applications , permet aux applications externeset aux clients de s'interfacer avec la blockchain.

8. Conclusion

Dans ce chapitre on a vu la technologie Blockchain et un besoin concret de sonutilisation ainsi que ces déférents types tell que privé et public et leur domaine d'application

Dans la partie Hyperledger et ces Projets on a vu le Hyperledger Fabric, ce model serale sujet à traiter dans le prochain chapitre ainsi la modélisation de notre projet.

Chapitre 3

Hyperledger Fabric et Modélisation

1. Introduction

Dans ce chapitre, nous donnons un aperçu du distributed ledger Hyperledger Fabric Plate-forme. Nous allons concentrer sur la version v2.3, ainsi la conception et l'analyse de notre projet. L'identification de la structure de l'information et des messages à utiliser
ainsi que les interactions qui pourraient être entre eux, les diagrammes de cas d'utilisation, lediagramme de classes, et le diagramme de séquences

2. Hyperledger Fabric

Hyperledger Fabric, projet open source de Linux Foundation, c'est le cadre modulaire et lanorme de facto pour les plateformes de blockchain d'entreprise. Destiné à développer des applications d'entreprise et des solutions sectorielles, l'architecture modulaire ouverte utilise des composants plug-and-play pour répondre à un large éventail de cas d'utilisation.

Avec plus de 120 000 organisations contributrices et plus de 15 000 d'ingénieurs contributeurs travaillant ensemble, Hyperledger Fabric offre une approche unique du consensus qui permet d'atteindre des performances à grande échelle tout en préservant laconfidentialité des données qu'exigent les entreprises.

IBM recommande aux entreprises de ne pas créer de solution de blockchain de productionen utilisant uniquement l'open source libre. IBM (et d'autres fournisseurs) propose des distributions commerciales incluant des outils et un support [(12)].

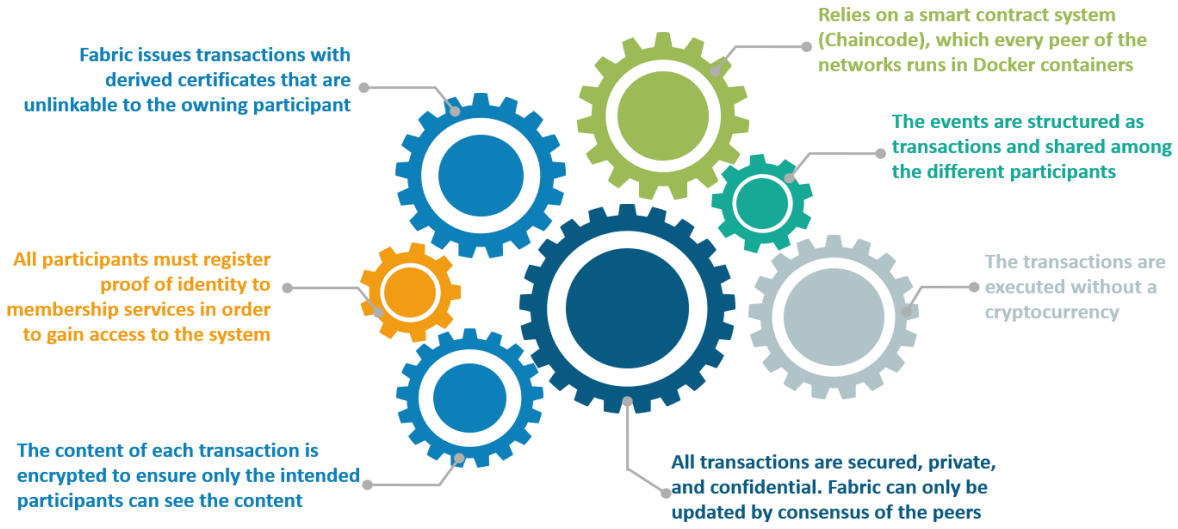
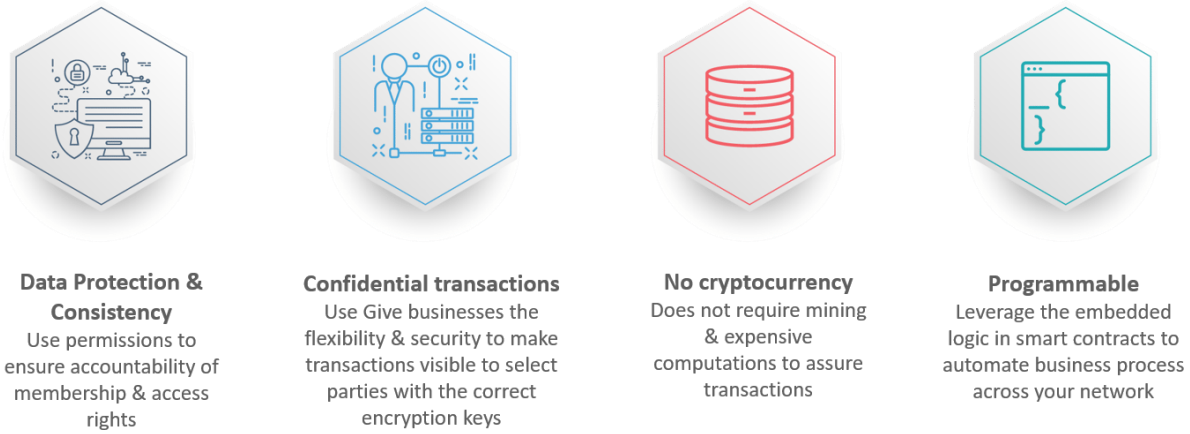


Figure 1 : Principe fonctionnalisées d'HyperLedger Fabric

3. Avantages de Hyperledger Fabric (12)



Réseau avec attribution de droits : Établissez une confiance décentralisée dans un réseaude participants connus plutôt qu'un réseau ouvert de participants anonymes.

Transactions confidentielles : Exposez uniquement les données à partager avec les partiesavec lesquelles vous souhaitez partager.

Architecture enfichable : Adaptez la blockchain aux besoins du secteur avec unearchitecture enfichable plutôt qu'une approche unique.

Démarrage simple : Programmez des contrats intelligents dans les langues dans lesquellesvotre équipe travaille aujourd'hui, au lieu d'apprendre des langues et des architectures personnalisées.

4. Fonctionnement HyperLedger Fabric [11]

Hyperledger Fabric est une plateforme d'entreprise ouverte, éprouvée et distribuée. La solution dispose de contrôles de confidentialité évolués de sorte que seules les données quevous souhaitez partager soient partagées entre les participants au réseau « autorisés » (connus).

Les contrats intelligents documentent les processus métiers que vous souhaitez automatiseravec des clauses auto-exécutables entre les parties écrites dans les lignes de code. Le code et les accords que la solution contient existent dans le réseau de blockchain décentralisé et distribué. Les transactions sont traçables et irréversibles, ce qui crée la confiance entre les organisations. Cela permet aux entreprises de prendre des décisions plus éclairées plusrapidement, et donc de gagner du temps, de réduire les coûts et diminuer les risques.

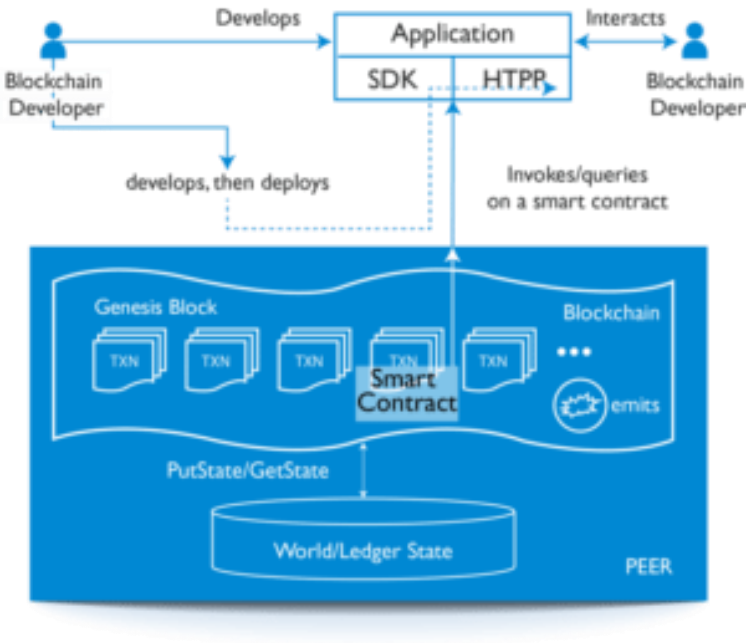


Figure 2 : architecture Hyperledger Fabrique [12]

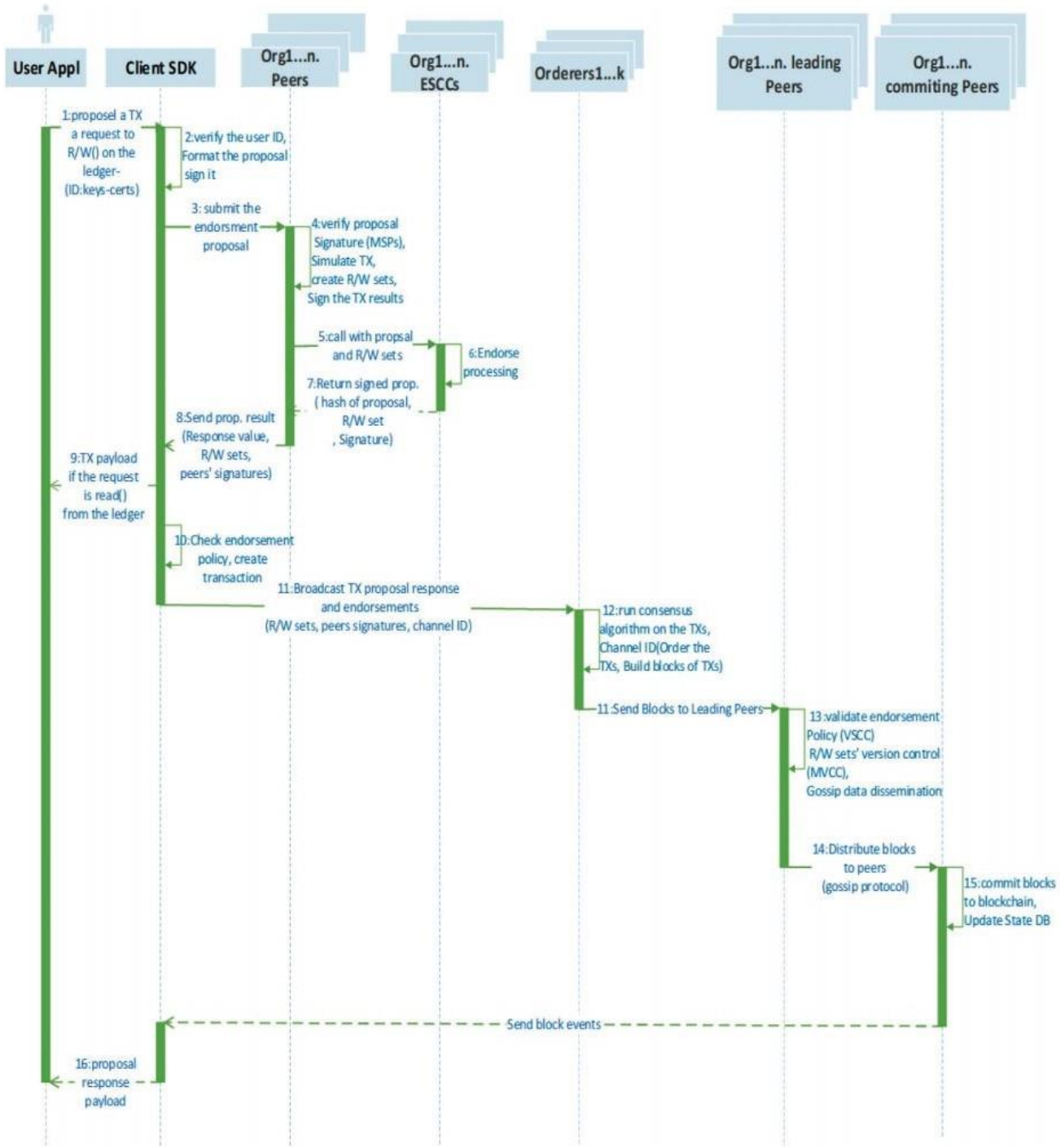


Figure 3 : Flux transaction dans HyperLedger Fabric [13]

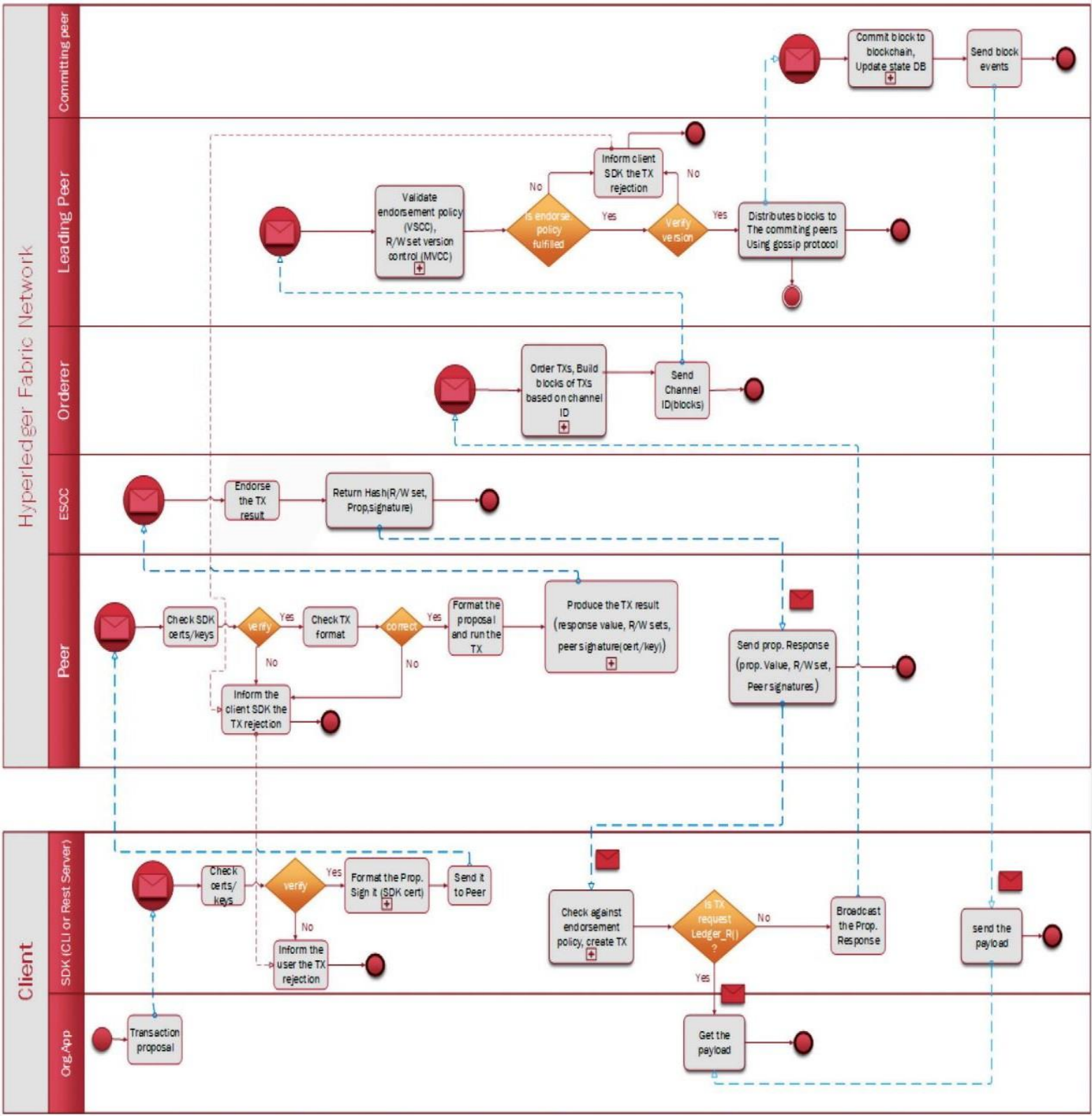


Figure 4 : Modèle de processus métier et notation [13]

5. Notions clés [14] :

5.1. Assets

Les assets peuvent aller du tangible (immobilier et matériel) à l'intangible (contrats et propriété intellectuelle). Hyperledger Fabric offre la possibilité de modifierles actifs à l'aide de transactions de code chaîne.

Les assets sont représentés dans Hyperledger Fabric sous la forme d'une collectionde paires clé-valeur, avec des changements d'état enregistrés en tant que transactions sur un grand livre de canal . Les actifs peuvent être représentés sous forme binaire et/ou JSON.

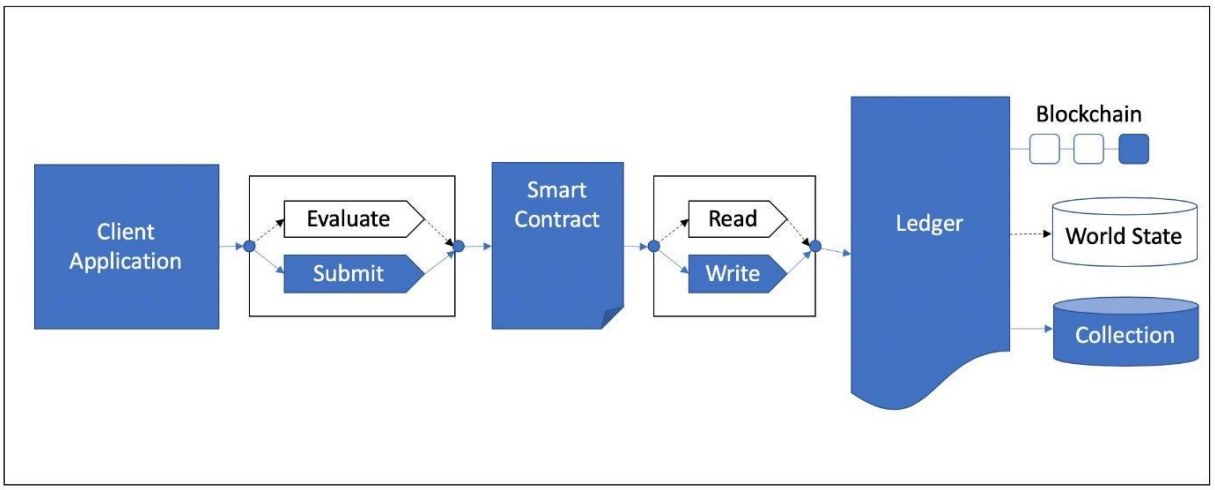


Figure 5 : Soumettre le chemin de la transaction

5.2. Ledger

Le ledger est l'enregistrement séquencé et inviolable de toutes les transitions d'état dans le Fabric. Les transitions d'état sont le résultat d'appels de code de chaîne (« transactions ») soumis par les parties participantes. Chaque transaction génère unensemble de paires clé-valeur d'actif qui sont validées dans le ledger en tant que créations, mises à jour ou suppressions.

Le ledger est composé d'une blockchain pour stocker l'enregistrement séquencé immuable en blocs, ainsi que d'une base de données d'état pour maintenir l'état actuel dela structure. Il y a un registre par canal. Chaque pair conserve une copie du registre pourchaque canal dont il est membre.

Quelques caractéristiques d'un ledger Fabric :

- Interrogez et mettez à jour le grand livre à l'aide de recherches basées sur des clés, derequêtes de plage et de requêtes de clé composite
- Requêtes en lecture seule utilisant un langage de requête riche (si vous utilisezCouchDB comme base de données d'état)
- Requêtes d'historique en lecture seule : interrogez l'historique du grand livre pourune clé, permettant des scénarios de provenance des données
- Les transactions se composent des versions des clés/valeurs qui ont été lues en chaincode (read set) et des clés/valeurs qui ont été écrites en chaincode (write set)
- Les transactions contiennent les signatures de chaque pair endosseur et sont soumisesau service de commande
- Les transactions sont ordonnées en blocs et sont « livrées » d'un service decommande à des pairs sur un canal
- Les pairs valident les transactions par rapport aux politiques d'approbation etappliquent les politiques
- Avant d'ajouter un bloc, une vérification de version est effectuée pour s'assurer queles états des actifs qui ont été lus n'ont pas changé depuis le temps d'exécution du code chaîne
- Il y a immuabilité une fois qu'une transaction est validée et engagée
- Le registre d'un canal contient un bloc de configuration définissant des politiques,des listes de contrôle d'accès et d'autres informations pertinentes
- Les canaux contiennent des instances de fournisseur de services d'adhésion permettant de dériver du matériel cryptographique de différentes autoritésde certification

5.3. Privacy

Hyperledger Fabric utilise un registre immuable sur une base par canal, ainsi qu'un code chaîne qui peut manipuler et modifier l'état actuel des actifs (c'est-à-dire mettre à jour les paires clé-valeur). Un grand livre existe dans le cadre d'un canal - il peutêtre partagé sur l'ensemble du réseau (en supposant que chaque participant fonctionne sur un canal commun) - ou il peut être privatisé pour n'inclure qu'un ensemble spécifique de participants.

5.4. Security & Membership Services

Hyperledger Fabric sous-tend un réseau transactionnel où tous les participants ont des identités connues. L'infrastructure à clé publique est utilisée pour générer des certificats cryptographiques liés aux organisations, aux composants réseau et aux utilisateurs finaux ou aux applications clientes. En conséquence, le contrôle d'accès aux données peut être manipulé et régi sur le réseau plus large et au niveau des canaux. Cettenotion « autorisée » d'Hyperledger Fabric, associée à l'existence et aux capacités des canaux, permet de répondre aux scénarios où la confidentialité et la confidentialité sont des préoccupations primordiales.

5.5. Contrats intelligents (Smart Contract) ou Chaincode

Les contrats intelligents Hyperledger Fabric sont écrits en code chaîne et sont invoqués par une application externe à la blockchain lorsque cette application doit interagir avec le grand livre. Dans la plupart des cas, le code chaîne n'interagit qu'avec lecomposant de base de données du grand livre, l'état du monde (en l'interrogeant, par exemple) et non avec le journal des transactions.

Chaincode peut être implémenté dans plusieurs langages de programmation. Actuellement, Go, Node.js et le code chaîne Java sont pris en charge.

Les smart contract ne sont pas seulement un mécanisme clé pour encapsuler les informations et les garder simples sur le réseau, ils peuvent également être écrits pour permettre aux participants d'exécuter automatiquement certains aspects des transactions.

Un contrat intelligent peut, par exemple, être écrit pour stipuler le coût d'expédition d'un article où les frais d'expédition changent en fonction de la rapidité avec laquelle l'article arrive. Avec les conditions convenues par les deux parties et écrites dansle grand livre, les fonds appropriés changent automatiquement de mains lorsque l'article est reçu.

5.6. Consensus

Dans la technologie des registres distribués, le consensus est récemment devenu synonyme d'un algorithme spécifique, au sein d'une même fonction. Cependant, le consensus englobe plus que le simple accord sur l'ordre des transactions, et cette différenciation est mise en évidence dans Hyperledger Fabric par son rôle fondamental dans l'ensemble du flux de transaction, de la proposition et de l'approbation à la commande, la validation et l'engagement. En résumé, le consensus est défini comme la vérification en boucle de l'exactitude d'un ensemble de transactions constituant un bloc.

Le consensus est finalement atteint lorsque l'ordre et les résultats des transactions d'un bloc ont satisfait aux vérifications des critères de politique explicites. Ces contrôles et équilibres ont lieu pendant le cycle de vie d'une transaction et incluent l'utilisation de politiques d'approbation pour dicter quels membres spécifiques doivent approuver une certaine classe de transaction, ainsi que des codes de chaîne système pour garantir que ces politiques sont appliquées et respectées. Avant l'engagement, les pairs utiliseront ces codes de chaîne système pour s'assurer que suffisamment d'approbations sont présentes et qu'elles proviennent des entités appropriées. De plus, une vérification de version aura lieu au cours de laquelle l'état actuel du grand livre est convenu ou consenti, avant quedes blocs contenant des transactions ne soient ajoutés au grand livre.

6. HyperLedger Fabric composants :

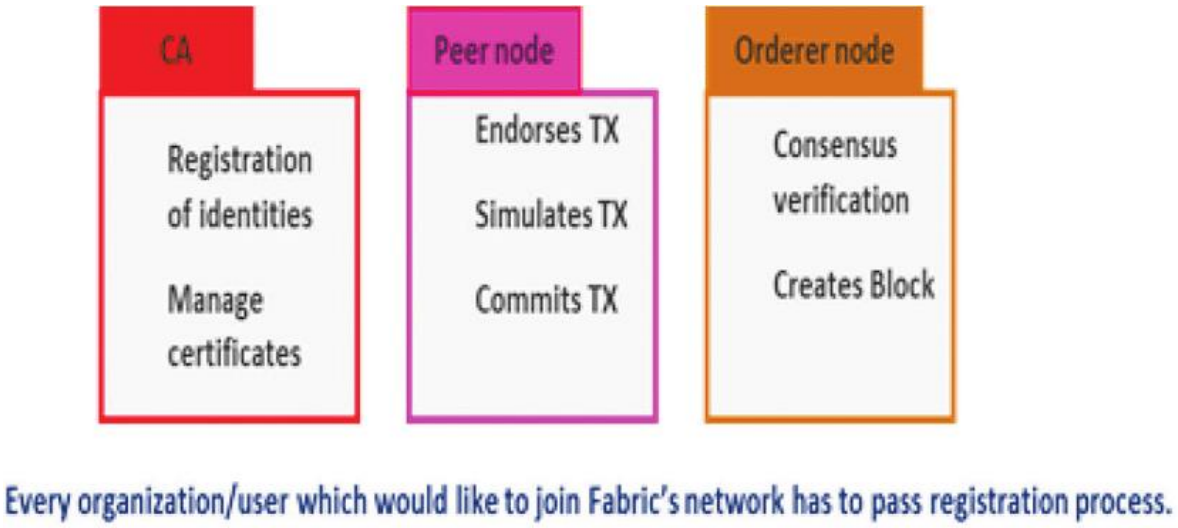


Figure 6 : HyperLedger Fabric composants [13]

6.1. Fonctionnalité des composants

Les composants de cette plate-forme coopèrent pour répondre aux exigences interorganisationnelles. Le réseau Hyperledger Fabric nécessite un intermédiaire, qui vatransférer ses données sur le réseau pour suivre le processus d'enregistrement avec une identité de confiance et de s'inscrire pour accéder au réseau Ressources. Les énoncéssuivants nous aident à comprendre ces composants fonctionnalités.

- Fabric-CA component : Chaque participant s'inscrit auprès de justificatif d'identitéaux services d'adhésion au réseau pour obtenir accès aux ressources du réseau.

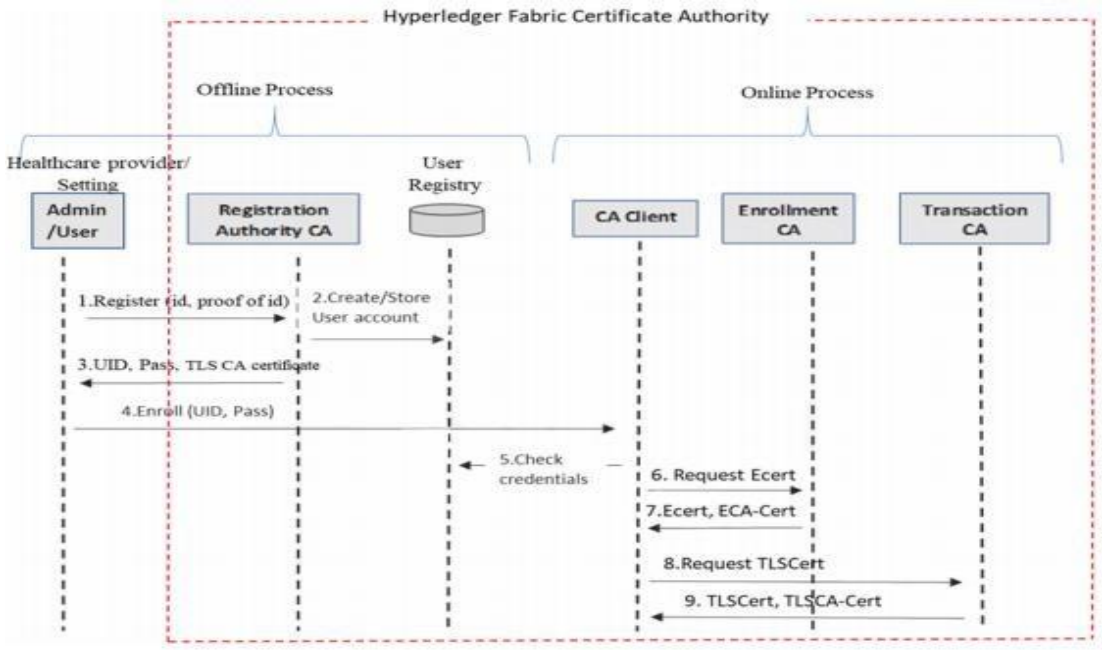


Figure 7 : Registration et enrôlement du membre, obtention de certificats.

- Membreship Service Provider : fournit des services de gestion identité. Tous lescomposants utilisent Membership Service Provider pour s'authentifier afin de communiquer et de partager des ressources. Figure (4)
- Peer node : le réseau blockchain est construit à partir des pairs détenues et contribuées par les différentes organisations. Un nœud peut rejoindre le réseau blockchain en exécutant une ou plusieurs fonctionnalités liées aux endosseurs, auxcommanditaires, aux validateurs et aux commiters.
- Endosseur node : simulera et signera ou rejettera les transactions. Il crée un ensemble de lecture/écriture de l'actif sur lequel une transaction a été demandée. Une transaction est une demande de lecture ou écrire sur le Ledger. Les nouvellestransactions sont annexées à la blockchain en tant que nouveau bloc et la base de données d'état mondial est mise à jour pour refléter le nouvel état.
- Orderer or ordering peer : exécute l'algorithme de consensus sur les transactions etordonner ces transactions de manière appropriée dans un bloc en fonction de leur ID de canal. Il envoie les blocs de transactions aux homologues validants/principaux.
- Validating peer : vérifie les transactions par rapport à l'approbation politique

(c'est-à-dire que n pairs qui approuvent doivent simuler et signer la transaction avant qu'elle ne soit remise au donneur d'ordre). Il effectue également un contrôlede version pour s'assurer qu'il existe des ensembles R/W () identiques pour tous les pairs. Ensuite, il envoie la transaction aux pairs qui s'engagent.

- Committing peer : recevez le bloc de transactions du pair validant/principal et l'ajoutez aux blocs et mettez à jour leur base de données respective (niveau DB ouCouchDB).
- Ledger : se compose de deux parties distinctes, bien que liées : la chaîne de blocset une base de données d'état des mots. Une blockchain est un immuable séquencede blocs, dont chacun contient un ensemble de transactions ordonnées. Un état mondial est une base de données qui contient les valeurs actuelles d'un ensemble d'états du grand livre.
- Channel : un sous-ensemble du réseau Fabric à travers lequel les pairs interagir lesuns avec les autres et avec les applications - un mécanisme par lesquels ces composants au sein d'un réseau blockchain peuvent communiquer et effectuer des transactions en privé.
- Smart contract : définit la logique exécutable qui génère de nouveaux faits qui sont ajoutés au Ledger. Pour mieux appréhender la notion d'interfonctionnalitéentre ces composants, la figure 3 illustre un diagramme de séquence de flux de transaction dans le réseau Hyperledger Fabric. Pour rendre le diagramme de séquence plus précis, nous n'avons pas inclus points de défaillance ou faussesconditions dans le réseau. La figure 4 représente un Business Process Model andnotation (BPMN) diagramme, qui présente une vue différente du flux de transaction à travers le réseau.

7. Travaux Connexes :

Les transactions commerciales entre les entreprises évolue avec le temps nécessite une sécurité et de confiance contre le piratage et la falsification des données, pour cela on a utilisé la technologie HyperLedger Fabric pour assurer les besoins mentionnés.

Identité : Acheteurs, vendeur.

Organisation : On a une seule organisation c'est notre site e-commerce qui permet à ces identités d'effectuer des achats vers elle.

8. Méthodologie

La figure 6 montre l'architecture du système et la figure 7 montre le diagramme de séquence de transaction exécuté dans le système que nous proposons.

La résolution des problèmes où il y avait des incohérences dans ledit commandes a nécessité de nombreuses interventions manuelles en raison du manque de réseau approprié pour gérer les litiges d'entiercement, la mise en œuvre du papier se concentre sur la création d'un réseau de blockchain pour stocker l'historique des transactions et des commandes des clients. Ceux-ci sont illustrés dans l'architecture du système ci-dessous.

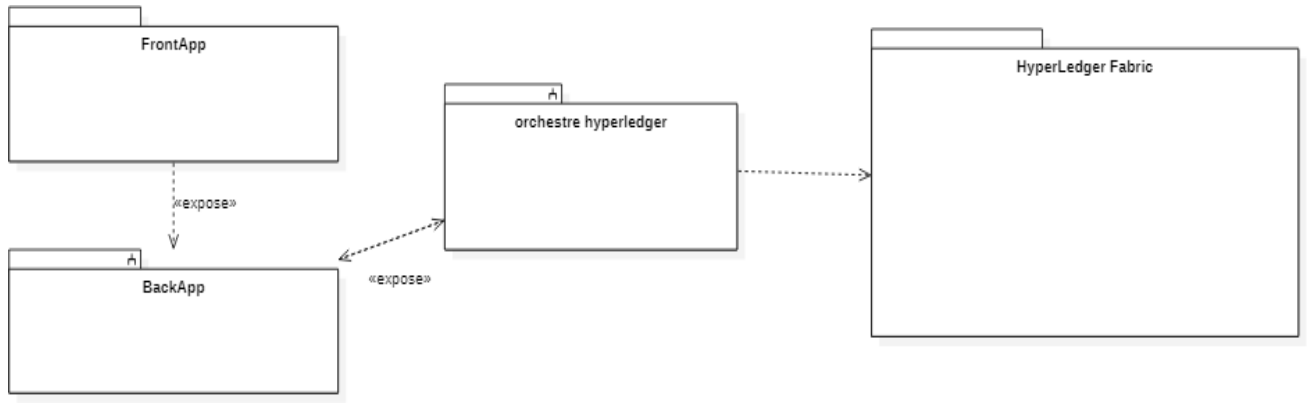


Figure 8 : Architecture du système

Quatre couches d'architecture fournissent l'environnement nécessaire à la mise en placedu marché. Le serveur Angular servira à exécuter l'application FrontEnd, le backend est déployé sur un serveur Tomcat ainsi que l'application Orchestre Hyperledger. La couche du milieu supervisera toutes les transactions via des contrats intelligents utilisant

des séquestres. Le réseau fonctionnera toujours complètement même si les nœuds abandonnent et quittent le réseau. La perte d'informations qui survient avec la perte de ces nœuds sera répliquéeà partir d'autres nœuds pour reconstruire la même intégrité de réseau. Le contrat intelligent résout les transactions et tout écart annulera la transaction et enverra des notifications en conséquence.

Les composants qui composent le marché décentralisé sont

1. Participants – Qui comprendront l'acheteur et le vendeur
2. Orders – Quel sera les transactions ou les achats
3. Transactions – Quels seront les contrats utilisés et leurs paramètres.

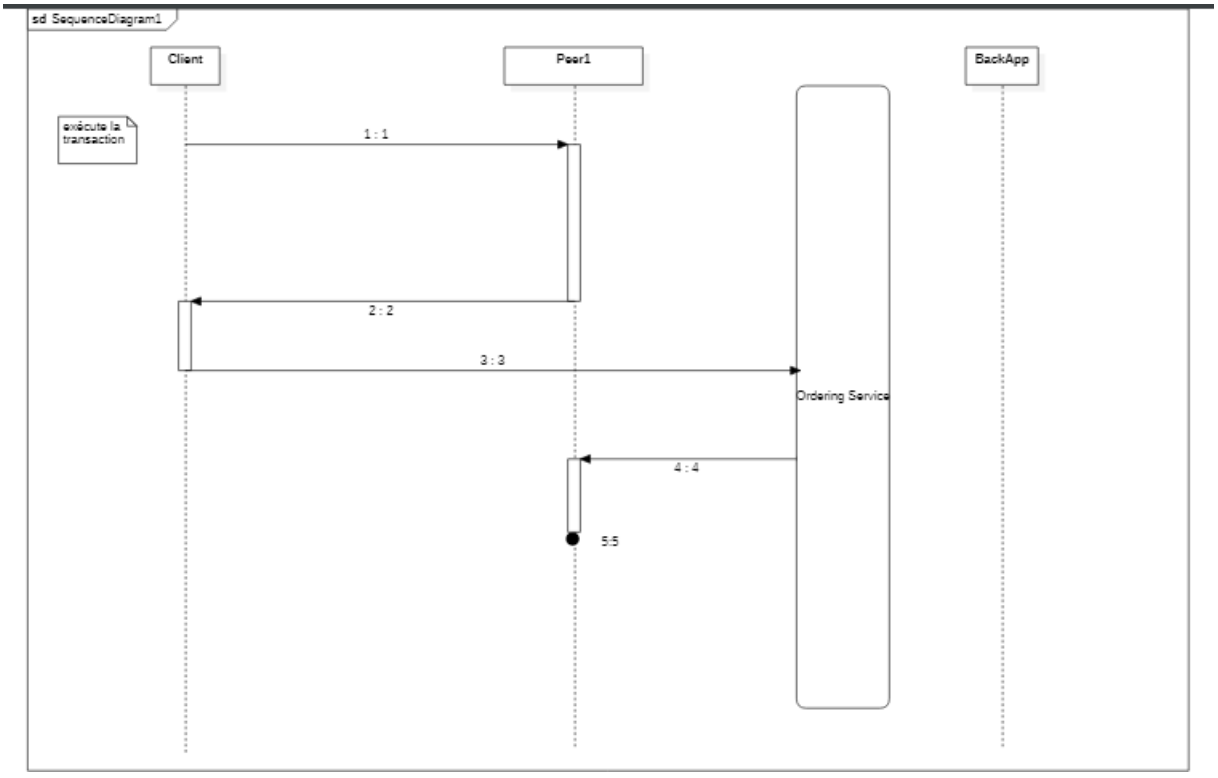


Figure 9 : Diagramme de séquence de transaction exécuté par l'orchestre Hyperledger1-

Proposition de transaction. Un client blockchain, qui représente une organisation,

crée une proposition de transaction et l'envoie à des pairs

d'approbation, comme défini dans la politique d'approbation. La proposition contient des informations concernant l'identité du proposant, lacharge utile de la transaction, un nonce et un identifiant de transaction.

2- Exécuter (endossement) : l'endossement consiste en la simulation de la transaction. Les endosseurs produisent un jeu d'écriture, contenant les cléset leurs valeurs modifiées, et un jeu de lecture. Les pairs d'endossement vérifient également l'exactitude de l'exécution de la transaction.
   L'endossement est envoyé en tant que réponse à la proposition et contient le jeu d'écriture, le jeu de lecture, l'ID de transaction, l'ID de l'endosseur etla signature de l'endosseur. Lorsque le client recueille suffisamment de recommandations (qui doivent avoir le même résultat d'exécution), il crée la transaction et l'envoie au service de commande. La phase de validation élimine tout éventuel non-déterminisme.

3- Vérifie si le client blockchain qui a soumis la proposition de transactiondispose des autorisations appropriées (autorisations de diffusion et de réception), sur un canal donné.

4- Valider. Tout d'abord, chaque pair valide les transactions reçues en vérifiant si une transaction suit la politique d'endossement correspondante.Après cela, une vérification de conflit en lecture-écriture est exécutée sur toutes les transactions du bloc, de manière séquentielle. Pour chaque transaction, il compare les versions des clés dans le jeu de lecture aveccelles actuellement sur le grand livre. Il vérifie si les valeurs sont les mêmes. Au cas où ils ne correspondent pas, les pairs rejettent la transaction.

5- Enfin, le ledger est mis à jour, dans lequel le ledger ajoute le bloc créé à sa tête. Le grand livre ajoute les résultats des contrôles de validité, y compris les transactions non valides.

9. Conclusion

Dans ce chapitre Nous avons présenté la solution HyperLedger Fabric, aussi l'objectifet l'intérêt de ce travail au début, et la modélisation.

Dans le chapitre suivant nous avons essayé de présenter les outils de développement de notre projet et les applications, les Langages de programmation, et enfin quelque interfacede site

Chapitre 4

Réalisations et Tests

1. Introduction

Ce quatrième chapitre sera consacré à la partie réalisation qui est une phase cruciale carelle représente le volet pratique de notre application. Dans cette ultime partie nous allons commencer par exposer les outils et l'environnement de développement ainsi que les techniques misent en œuvre pour atteindre les objectifs fixés dans les chapitres précédents. Par la suite nous allons montrer un aperçu des interfaces est des parties de code les plus importantes de notre projet.

2. Environnement de Programmation et Bibliothèques

Cette section va nous permettre d'énumérer les différentes technologies utilisées pour la mise en œuvre de notre système :

- Git [15]

C'est un logiciel de gestion de versions décentralisé. C'est un logiciel libre créé par Linus Torvalds, auteur du noyau Linux, et distribué selon les termes de la licence publique générale GNU version 2. Le principal contributeur actuel de git et depuis plus de 16 ans est Junio C Hamano. En 2016, il s'agit du logiciel de gestion de versions le plus populaire qui est utilisé par plus de douze millions de personnes.

- IntelliJ IDEA [15]

IntelliJ IDEA également appelé « IntelliJ », « IDEA » ou « IDJ » est un environnement de développement intégré (en anglais Integrated Development Environment - IDE) destiné au développement de logiciels informatiques reposant sur la technologie Java. Il est développé par JetBrains (anciennement « IntelliJ ») et disponible en deux versions, l'une communautaire, open source, sous licence Apache 2 et l'autre propriétaire, protégée par une licence commerciale. Tous deux supportent les langages de programmation Java, Kotlin, Groovy et Scala.

- Docker [15]

Docker est un logiciel libre permettant de lancer des applications dans des conteneurs logiciels. Selon la firme de recherche sur l'industrie, « Docker est un outil qui peut empaqueter une application et ses dépendances dans un conteneur isolé, qui pourra être exécuté sur n'importe quel serveur ». Il ne s'agit pas de virtualisation, mais de conteneurisation, une forme plus légère qui s'appuie sur certaines parties de la machine hôte pour son fonctionnement. Cette approche permet d'accroître la flexibilité et la portabilité d'exécution d'une application, laquelle va pouvoir tourner de façon fiable et prévisible sur une grande variété de machines hôtes, que ce soit sur la machine locale, un cloud privé ou public, une machine nue, etc.

Techniquement, Docker étend le format de conteneur Linux standard, LXC, avec une API de haut niveau fournissant une solution pratique de virtualisation qui exécute les processus de façon isolée. Pour arriver à ses fins, Docker utilise entre autres LXC, cgroups et le noyau Linux lui-même. Contrairement aux machines virtuelles traditionnelles, un conteneur Docker n'inclut pas de système d'exploitation, mais s'appuie au contraire sur les fonctionnalités du système d'exploitation fournies par la machine hôte.

La technologie de conteneur de Docker peut être utilisée pour étendre des systèmes distribués de façon qu'ils s'exécutent de manière autonome depuis une seule machine physique ou une seule instance par nœud. Cela permet aux nœuds d'être déployés au fur et à mesure que les ressources sont disponibles, offrant un déploiement transparent et similaire aux PaaS pour des systèmes comme Apache Cassandra, Riak ou d'autres systèmes distribués.

- Java (langage) [15]

Java est un langage de programmation orienté objet créé par James Gosling et Patrick Naughton, employés de Sun Microsystems, avec le soutien de Bill Joy (cofondateur de Sun Microsystems en 1982), présenté officiellement le 23 mai 1995 au *SunWorld*.

La société Sun a été ensuite rachetée en 2009 par la société Oracle qui détient et maintient désormais Java.

Une particularité de Java est que les logiciels écrits dans ce langage sont compilés versune représentation binaire intermédiaire qui peut être exécutée dans une machine virtuelle Java (JVM) en faisant abstraction du système d'exploitation.

- Spring Boot [15]

Spring Boot est un framework qui facilite le développement d'applications fondées sur Spring en offrant des outils permettant d'obtenir une application packagée en *jar* , totalement autonome. Ce qui nous intéresse particulièrement, puisque nous essayons de développer des Microservices

- Hibernate [15]

C'est un framework open source gérant la persistance des objets en base de donnéesrelationnelle.

Hibernate est adaptable en termes d'architecture, il peut donc être utilisé aussi bien dans un développement client lourd, que dans un environnement web léger de type ApacheTomcat ou dans un environnement Java EE complet : WebSphere, JBoss ApplicationServer et Oracle WebLogic Server.

Hibernate apporte une solution aux problèmes d'adaptation entre le paradigme objet et les SGBD en remplaçant les accès à la base de données par des appels à des méthodes objet de haut niveau.

- Les services REST [15]

Représentent un style d'architecture pour développer des services web. Une API qui respecte les principes REST est appelée API-RESTful.
Les principes clés de REST impliquent la séparation de l'API en ressources logiques. Ce quirevient à penser à comment obtenir chaque ressource.
Une ressource est un objet ou une représentation d'objets contenant éventuellement des données. La manipulation de ces ressources repose sur le protocole HTTP à travers les méthodes d'actions GET, POST, PUT,PATCH, DELETE…etc.

- Ubuntu (système d'exploitation) [15]

Ubuntu est un système d'exploitation GNU/Linux basé sur Debian. Il est développé, commercialisé et maintenu pour les ordinateurs individuels (desktop), les serveurs (Server) et les objets connectés (Core) par la société Canonical.

- JDBC (Java Database Connectivity) [15]

Est un ensemble de classes Java qui permet de se connecter à une base de données distante sur le réseau, Ce paquetage permet de formuler et gérer les requêtes aux bases de données relationnelles. Pour travailler avec un SGBD il faudrait disposer des classes drivers qui implémentent les interfaces JDBC. IV.1.6.SQL (Structured Query Language) SQL est lelangage des bases de données relationnelles répondant à la fois à la problématique de création des objets de bases de données (modèle), de manipulation des données (algèbre relationnelle), de gestion de la sécurité (« droits d'accès »), de traitements locaux de données (procédures). De plus, il est désormais doté d'extensions objet.

- Angular [15]

C'est un cadriciel (framework) côté client, open source, basé sur TypeScript, et co- dirigé par l'équipe du projet « Angular » à Google et par une communauté de particuliers et de sociétés. Il permet la création d'applications Web et plus particulièrement de ce qu'on appelle des « *Single Page Applications* » : des applications web accessibles via une page web unique qui permet de fluidifier l'expérience utilisateur et d'éviter les chargements de pages à chaque nouvelle action. Le Framework est basé sur une architecture du type MVC etpermet donc de séparer les données, le visuel et les actions pour une meilleure gestion des responsabilités. Un type d'architecture qui a largement fait ses preuves et qui permet une forte maintenabilité et une amélioration du travail collaboratif.

- Bootstrap [15]

C'est une collection d'outils utiles à la création du design (graphisme, animation et interactions avec la page dans le navigateur, etc.) de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option.

C'est l'un des projets les plus populaires sur la plate-forme de gestion de développement GitHub.

- Test network [14]

Le Fabric test network dans le référentiel d'exemples fournit un réseau de test basé sur Docker Compose avec deux homologues d'organisation et un nœud de service de commande. Vous pouvez l'utiliser sur votre ordinateur local pour exécuter les exemples répertoriés ci-dessous. Vous pouvez également l'utiliser pour déployer et tester vos propres codes de chaîne et applications Fabric.

3. Architecture de projet

Apres avoir exposé l'ossature de la partie données de notre projet, nous allons épiloguer l'architecture de notre projet en termes de déploiement des modules qui la composent sur le différent terminaux (Ordinateur, serveur) ainsi nous terminerons par une présentation de la structure du code source des applications.

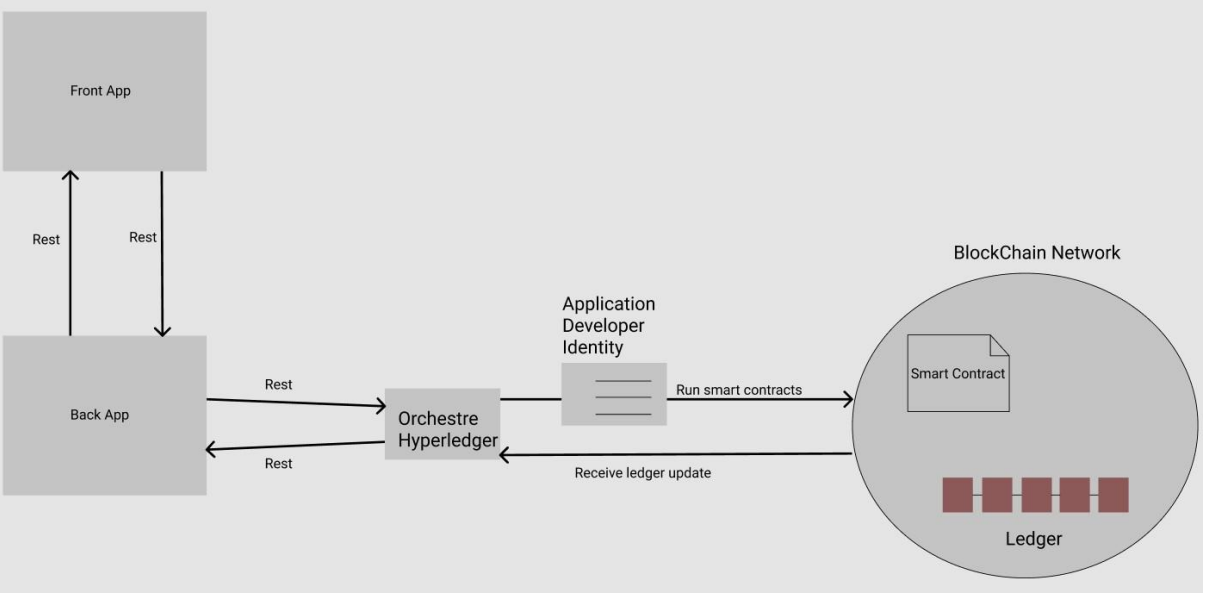


Figure 1 : Architecture du projet FabMarket

4. Projet FabMarket

Pour réaliser notre projet basé sur sérialisation des données avec la technologie HyperLedgerFabric, on a créé 3 applications déférents FrontEnd en Angular,et le rest en Java déployé sur des serveurs Tomcat BackEnd et l'orchestre HyeperLedger, la chainCode est une smart contractqu'on a développé en java aussi.
Les 3 premiers applications sont des applications séparées, ce veux dire que le front est le back chaqu'un entre eux est indépendant communique entre eux via des Web Services utilisant l'architecture REST.

4.1. FrontApp :

4.1.1. Structure du code source

Nous avons organisé selon le concept du design pattern « MVC » qui sépare le code entrois niveaux, on définit le rôle de chaque fichier dans l'application :

- Modèle : cette partie gère les données de notre application. Elle contient les 2 parties :
  - Service : Didier à la récupération des données depuis le BackApp
  - Model : elle contient les Objets qu'on a utiliser le moment de récupération ou traitement des données.
- Vue : cette partie entièrement programmée en HTML et les balises Angular pour rendre notre Html dynamique à l'aide aussi du Framework Bootstrap v4, assure l'interaction entre les utilisateurs et la machine autrement dit c'est la partie visible de l'application oufrontend.
- Contrôleur : cette partie gère la logique du code qui prend des décisions. C'est en quelque sorte l'intermédiaire entre le modèle et la vue, son codage est du pure TypeScripte.

4.1.2. Choix du langage :

Pour l'implémentation de cette application on a utilisé le langage de programmationTypeScript et le framework Angular version 8.

4.2. Back App

C'est une application backend de notre projet, son rôle principal est d'expose des WS est enregistre les informations dédier au BDD SQL via des requête HQL et faire appel au l'application Orchestre Hyperledger pour exécuter des transactions vers notre Ledger.

4.2.1. BDD SQL

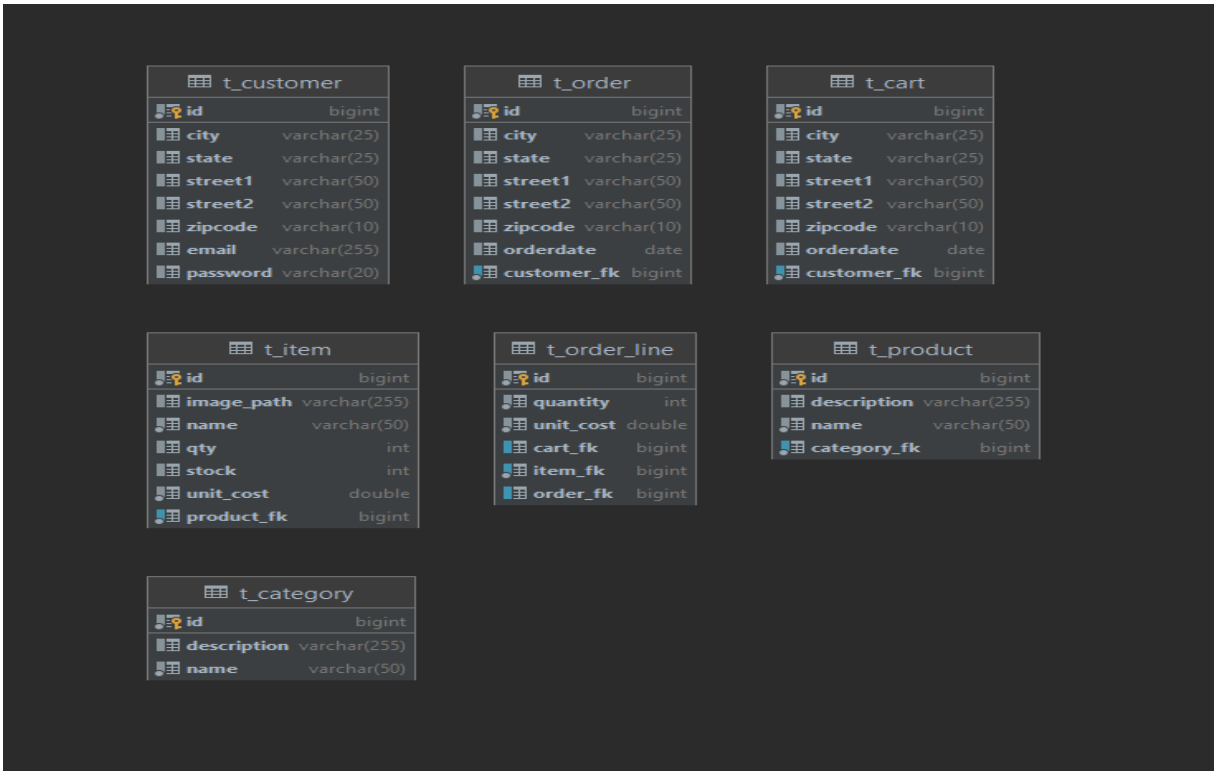


Figure 2 : Schéma physique de la base de données

4.2.2. Structure du code source

On a utilisé que le model et le contrôleur, la vue sera ce que l'application FrontAppconsommé.

- Modèle : cette partie gère les données de notre application. Elle contient les 3 parties :
  - DAO (Data Access Object) : Didier à la récupération des données depuis la BDD,elle est développée à l'aide du framework Hibernate en JAVA.

- Model : elle contient les POJO qu'on a utilisé le moment de récupération outraitement des données.
- Service : contient les class dédier au traitement des données

- Contrôleur : cette partie gère la logique du code qui prend des décisions. C'est en quelquesorte l'intermédiaire entre le modèle et la vue, son codage est du pure TypeScripte.

4.3. Orchestre Hyperledger

C'est l'application qui orchestre les échanges entre le BackApp et l'Hyperledger Fabric.

Elle est bien en design pattern MVC aussi, le front est les informations qu'on expose via les WS, le rôle de cette application est d'enrôlé les utilisateurs et les admins qui peuvent réagir avec notre dernier application Ledger Fabric via des transactions.

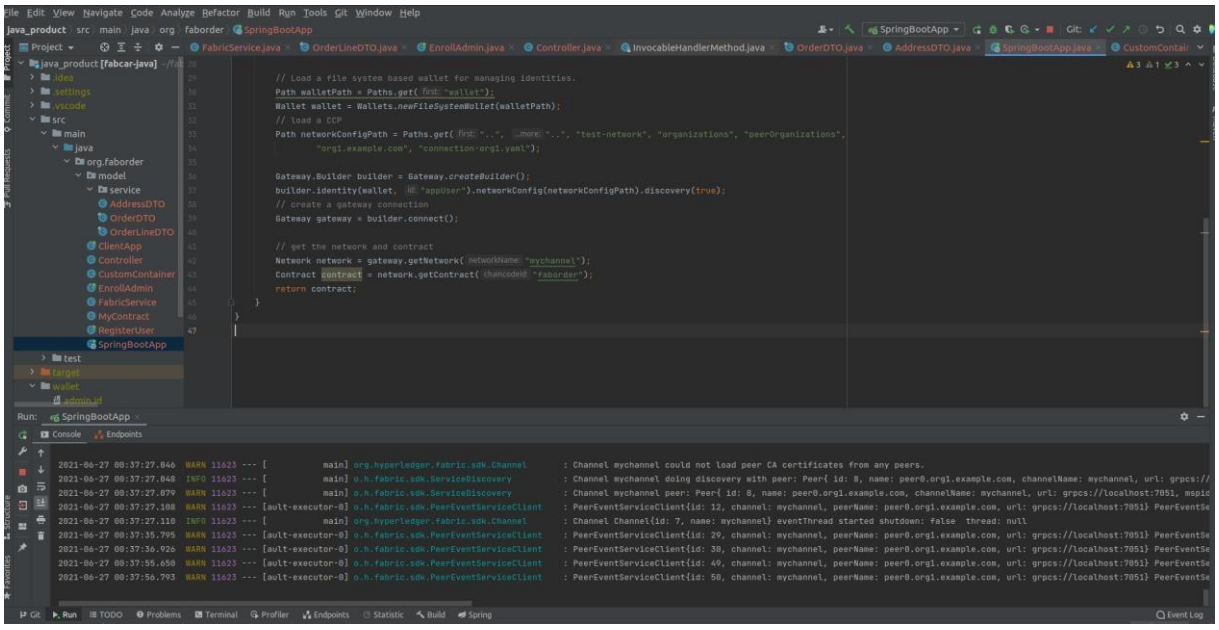


Figure 3 : Capture code source Orchestre Hyperledger

4.4. Test Network [14]

Le réseau de test se compose de deux organisations homologues et d'une organisation de commande. Les deux organisations homologues exploitent chacune un homologue, tandis que l'organisation commanditaire exploite un service de commande Raft à nœud unique. Nous utiliserons également le réseau de test pour créer un canal unique nommé mychannel dont les deux organisations homologues seront membres.

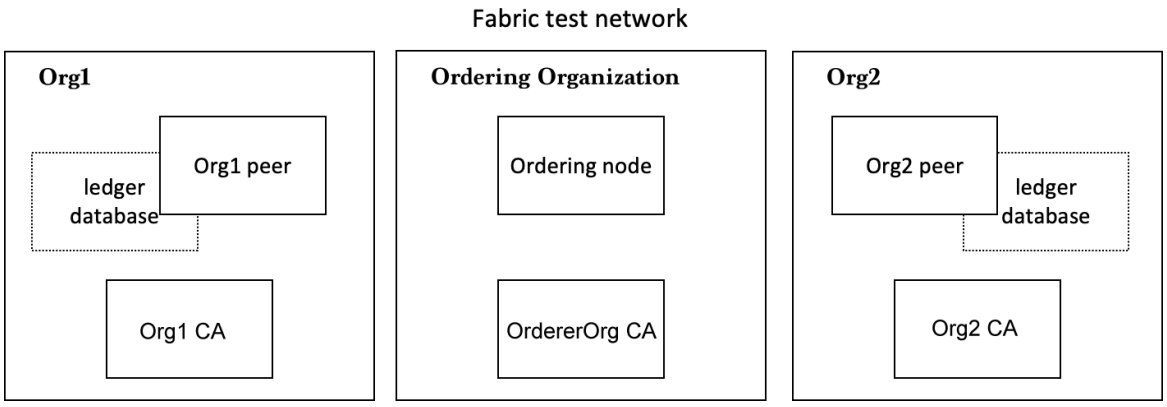


Figure 4 : Architecture du réseau test network

Pour lancer le test network on lance un scripte annoté, ./startNetwork.sh, qui met en place un réseau Fabric à l'aide des images Docker sur notre ordinateur local.

L'exécution de ce scripte crée un réseau Fabric composé de deux nœuds homologues,un nœud de commande.

Après le lancement du réseau on doit créer une chaine que les utilisateurs qui ontun certificat d'autorité Fabric-CA, par l'exécution de la commande

./network.sh createChannel

Maintenant c'est le temps de déployement de notre smart contract (chaine code)qu'on a développésur le réseau et particulièrement accessible que via un accée vers notrechaine par la commande suivante :

```
./network.sh deployCC -ccn {{nom_chaine}} -ccp
{{chemain_vers_notre_chainecode}} -ccl java
```

La figure suivante est une capture d'écran de l'IDE Visuel Code qui nous a offrir une possibilité de visualiser les différents composants docker (Images, Container, Volumes …) générer le moment de déploiement du Contract.

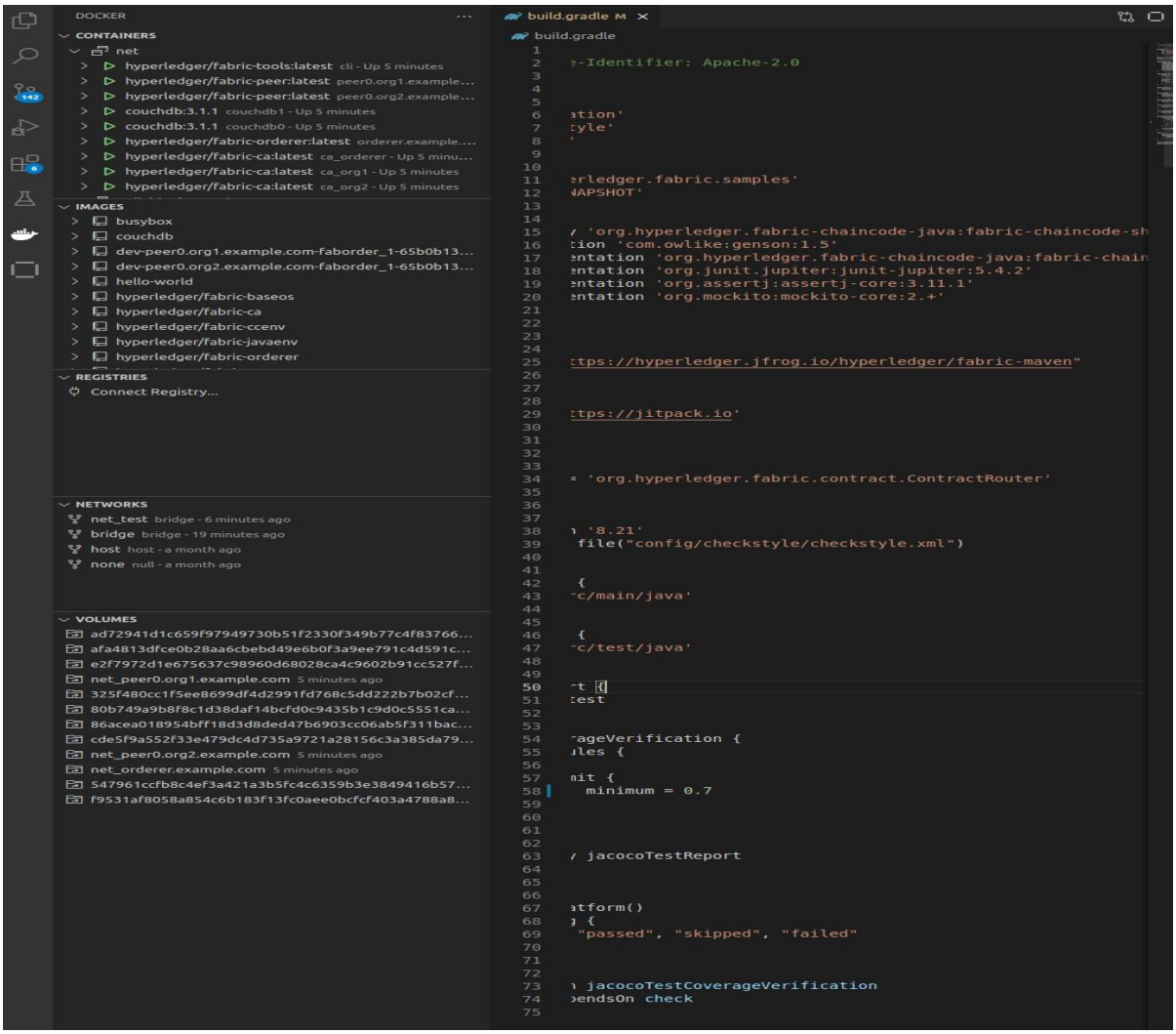


Figure 5 : déférents composant générer par la solution Docker

4.4.1. Chaine Code « Fabric Order » :

Notre smart contract est intituler « FABORDER », C'est une application qu'on a développé en java qui nous permet de créer et de transférer un actif en plaçant des commandes dans le grand livre et en les récupérant.

4.4.2. Structure du code source

Notre Chaine Code ou smart contract se compose de deux class en Java,

- La première classe contient les informations de notre objet.

```
@DataType()
public final class Order {

   @Property()
   private final String details;

   public String getDetails() {
      return details;
   }

   public Order(@JsonProperty("details") final String details) {
      this.details = details;
   }
…
…
…
```

- La deuxième classe contient les définitions des transactions autorisées à être communiqué par les paires déployé sur la chaine.

```
@Contract(
      name = "FabOrder",
      info = @Info(
            title = "FabOrder contract",
            description = "The hyperlegendary order contract",
            version = "0.0.1-SNAPSHOT",
            license = @License(
                  name = "Apache 2.0 License",
                  url = "http://www.apache.org/licenses/LICENSE-2.0.html"),
            contact = @Contact(
                  email = "f.orderr@example.com",
                  name = "F Orderr",
                  url = "https://hyperledger.example.com")))
@Default
```

```
public final class FabOrder implements ContractInterface {

    private final Genson genson = new Genson();

    private enum FabOrderErrors {
        ORDER_NOT_FOUND,
        ORDER_ALREADY_EXISTS
    }

    /**
     * Retrieves a order with the specified key from the ledger.
     *
     * @param ctx the transaction context
     * @param key the key
     * @return the Order found on the ledger if there was one
     */
    @Transaction()
    public Order queryOrder(final Context ctx, final String key) {
        ChaincodeStub stub = ctx.getStub();
        String orderState = stub.getStringState(key);

        if (orderState.isEmpty()) {
            String errorMessage = String.format("Order %s does not exist", key);
            System.out.println(errorMessage);
            throw                    new
                                     ChaincodeException(errorMessage,
FabOrderErrors.ORDER_NOT_FOUND.toString());
        }

        Order order = genson.deserialize(orderState, Order.class);

        return order;
    }
```

5. Aspect graphique de l'application :

5.1. Home page

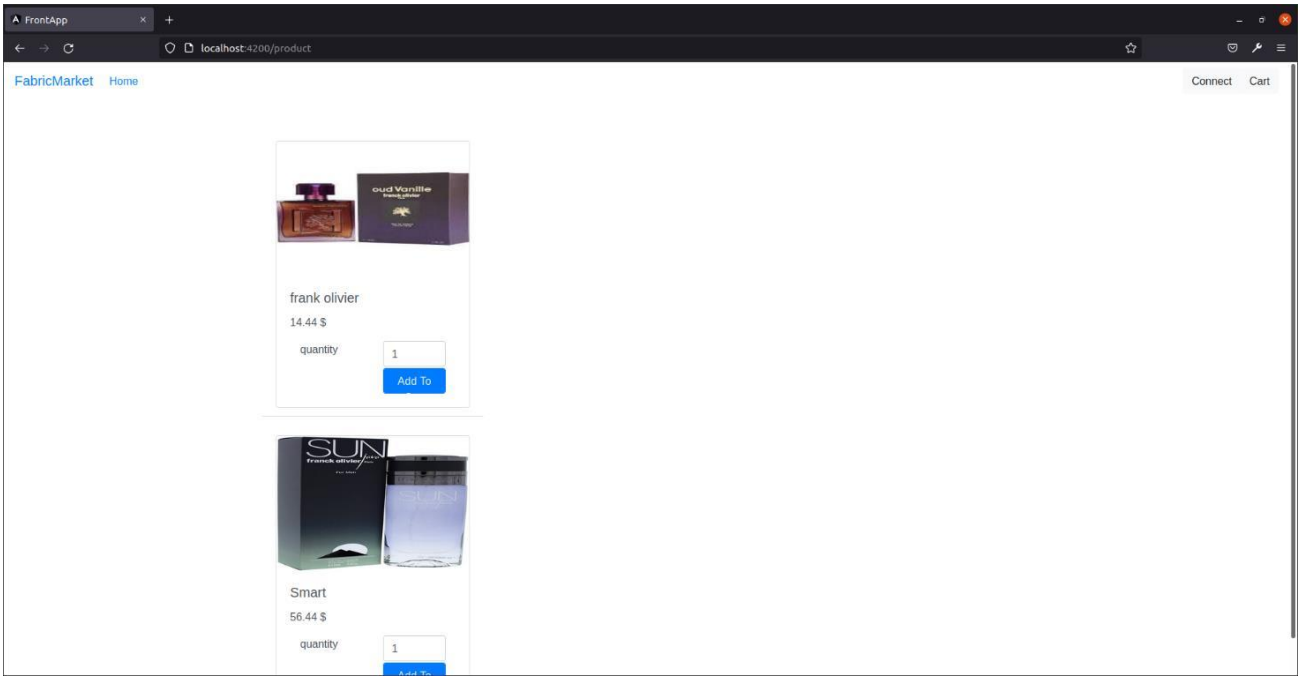


Figure 6 : Home page

5.2. Cart page

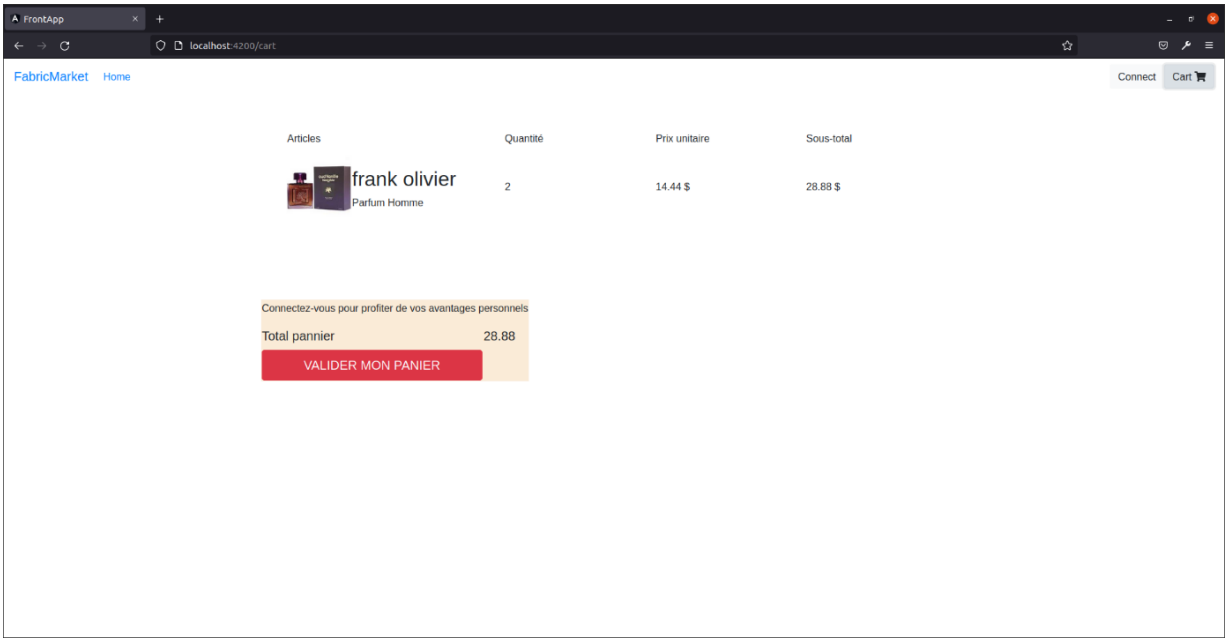


Figure 7 : Panier page

5.3. Connection page

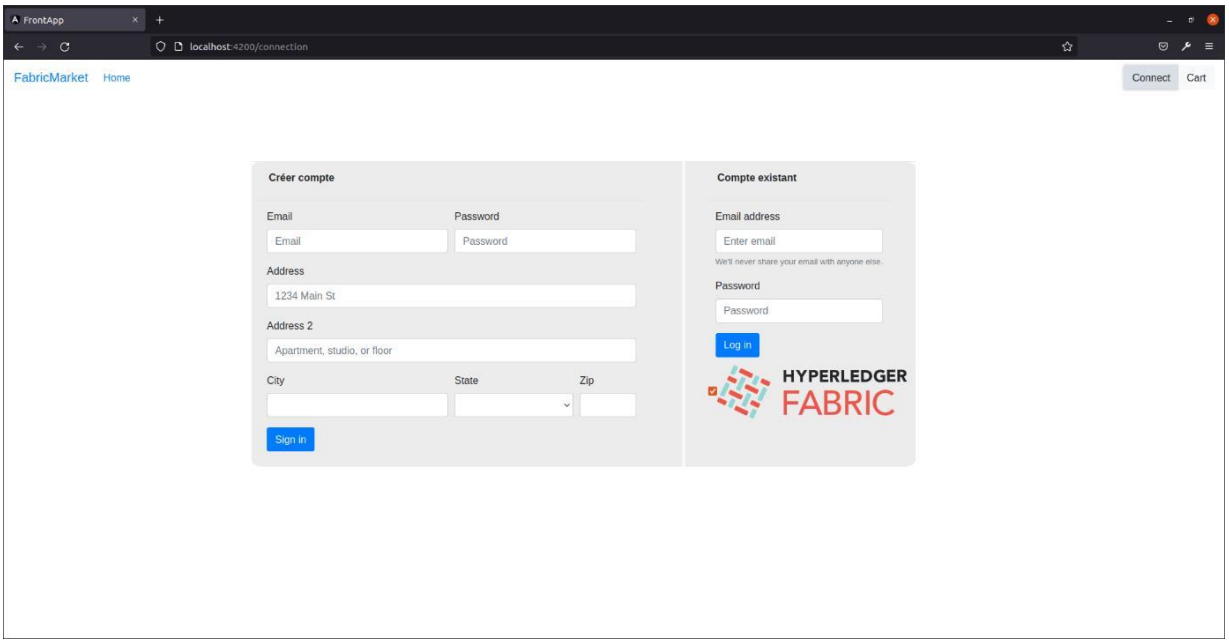


Figure 8 : Connection page

6. Conclusion

Dans ce chapitre on a vu les déférents technologies, solutions et Framework qu'on a utilisé pour développer nos applications.

Finalement, je suis très convaincu de travail que j'ai développé. Une application basée sur la technologie blockchain, pour l'intérêt d'enregistrer l'historique des commandes passer sur un site web e-commerce dans le ledger fournie par la solution HyperLedger Fabric.

Conclusion Générale

Le système proposé dans ce mémoire, FabMarket, est l'une des nombreuses applications pour lesquelles la blockchain pourrait bientôt être utiliser. Cependant, les scénarios appliqués sont limités et nécessitent d'être affinés s'ils doivent être appliqués à plus grande échelle. La validation et l'historique des transactions fourniront également de nouveaux moyens de données à utiliser pour améliorer les systèmes actuels, par exemple les utilisateurs authentiques avec l'achat et la propriété actuelle du produit sont les seuls à avoir la permission d'effectuer des achats, L'idée d'un réseau entre les réseaux avec la transparence et l'équité à la base remplacera la pensée traditionnelle des modèles de transaction car l'open source a si souvent vu que les systèmes propriétaires finiront par prendre quelques pas derrière leurs alternatives open source. Le compositeur de Fabric hyperledger a fourni le bloc de construction fondamental de notre blockchain où aucun utilisateur ne peut manipuler le contenu d'un bloc et le réintroduire avec succès, cette nature nous a permis de créer beaucoup plus de fonctionnalités rapidement sans obstacles de mise en œuvre de niveau inférieur qui étaient un problème majeur au départ. La technologie Blockchain progresse à un rythme qui a attiré l'attention de tous dans l'industrie et son application n'est limitée que par notre imagination où il pourrait y avoir une application si révolutionnaire qu'elle pourrait bien changer le visage de la transaction et créer de nouvelles industries du futur.

Références

1. International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181

2. Polycopié du cours E-commerce 2020-2021 Nassim Dennouni, University of Chlef Hassiba Benbouali

3. E-Commerce au Maroc : réalités et perspectives, Bouchra JEGHAOUI 2003

4. Blockchain privée ou publique Sébastien Bourguignon 2018,

5. https://blockchainfrance.net/decouvrir-la-blockchain/c-est-quoi-la-blockchain/

6. Building Blockchain Projects
Narayan Prusty ISBN: 978-1-78712-214-7

7. Mastering Blockchain – Third Edition
Imran Bashir ISBN: 978-1-83921-319-9

8. Article par Ludovic Lars le 20 juin 2018

9. IoT-Blockchain Enabled Optimized Provenance System for Food Industry 4.0 Using Advanced Deep Learning May 2020 Authors: Prince Waqas Khan, Yungcheol Byun DOI:10.3390/s201029903

10. https://www.hyperledger.org/

11. Securing Blockchain Networks like Ethereum and Hyperledger Fabric Auteur(s): Parisi, Alessandro, Editeur: Packt Publishing, Année de Publication: 2020 ISBN: 978-1-83864-648-6

12. Blockchain with Hyperledger Fabric

Auteur(s): Gaur, Nitin O'dowd, Anthony Novotny, Petr, Editeur: Packt Publishing, Ann é dePublication: 2020 pages: 757 ISBN: 978-1-83921-875-0

13. A Secure and Scalable Data Source for Emergency Medical Care using Blockchain Technology, vol 7, 2019 ISSN 2211-7938; eISSN 2211-7946

14. https://hyperledger-fabric.readthedocs.io/en/release-2.3/

15 https://fr.wikipedia.org/