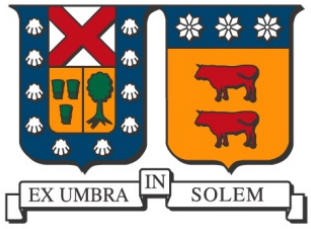
UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE INFORMÁTICA
SANTIAGO - CHILE



“BLOCKCHAIN APLICADO A TRAZABILIDAD EN RECICLAJE DE RESIDUOS ORGÁNICOS EN EDIFICIOS.”

IGNACIO ALEJANDRO NORAMBUENA ACUÑA

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL EN INFORMÁTICA

Profesor Guía: Hernán Astudillo Rojas
Profesor Correferente: Cristián Orellana

Octubre - 2019

DEDICATORIA

Dedicado a todos aquellos que con compañía y conversación me dieron su apoyo en todo este tiempo.

RESUMEN

Resumen— La trazabilidad de activos es valorada en diversas organizaciones. En reciclaje, permite acceder a nuevos modelos de negocio y es vital para cumplir nuevas normativas como la ley REP. Por otro lado, blockchain surge como una tecnología capaz de dar transparencia e inmutabilidad a sistemas transaccionales contruidos sobre él.

Este documento busca dilucidar si un sistema de trazabilidad basado en blockchain es una alternativa para lograr un proceso de reciclaje de residuos confiable y transparente.

Tomando como referencia un sistema de reciclaje existente, se propuso una versión basada en blockchain y se compararon sus ventajas y desventajas. Como resultado se obtuvo que este no es un buen caso de uso, sin embargo se sientan las bases para comprender cuales son los mejores casos de uso para esta tecnología.

Palabras Clave— Blockchain; Ledger; Trazabilidad; Ethereum; Hyperledger;

ABSTRACT

***Abstract*—** Assets traceability it's important in many organizations. Applied in recycling, allows to get access to new business models and be aligned with the new legislation, REP law. On the other hand, blockchain arises as a technology capable to give confidence and inmutability to transactional systems build over it.

The objective of this document it's unveils if a traceablility system based in blockchain it's an alternative to achieve a trustworthly and transparent waste recycling process.

Based in an existing recycling system a blockchain version was proposed and compared with the original, getting its advantages and disadvantages. As result, blockchain is not a good alternative for this system, however the results are important to understand the correct use cases for this technology.

***Keywords*—** Blockchain; Ledger; Tracking; Ethereum; Hyperledger;

GLOSARIO

DI: Departamento de Informática.
UTFSM: Universidad Técnica Federico Santa María.

ÍNDICE DE CONTENIDOS

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

INTRODUCCIÓN

La trazabilidad de activos toma cada día mas importancia en diversas industrias. Ha pasado de ser un valor agregado capaz de entregar transparencia en procesos, mejores controles y reducción riesgo, a ser una exigencia para diversas industrias en muchos países.

Implementar esto exige a las organizaciones sistemas que ayuden a cumplir este objetivo, arrastrando con ello una serie de desafios. Muchos de ellos estan relacionados con que los productos son extraidos, procesados y vendidos en muchos lugares, incluso en organizaciones diferentes.

Datos faltantes y desactualizados, distintas formas de almacenaje y formato de datos, propiedad de los datos y riesgo de falsificación son algunos de los problemas y desafios a los que se enfrenta generar sistemas de trazabilidad en muchas industrias.

Por otro lado, blockchain ha emergido en el último tiempo como una tecnología capaz de dar soporte a una nueva forma de almacenar y distribuir datos. Se trata de una base de datos distribuida donde los registros son almacenados con reglas claras y auditadas por toda la red en una estructura de datos virtualmente inmodificacble.

Así, un sistema basado en blockchain promete generar confianza en la informacion que se extrae de ella y homologar una sola forma de manejar los datos en diversas organizaciones, eliminando la desconfianza y disminuyendo el riesgo de errores en el manejo de los datos. Es por las características anteriores que blockchain surge como una alternativa interesante a tener en cuenta para crear sistemas de trazabilidad.

Esta memoria tiene como objetivo dilucidar si blockchain es una buena alternativa para un sistema de trazabilidad de residuos orgánicos en edificios y para ello se realizó una prueba de concepto del sistema basada en esta tecnología. A partir de lo aprendido se propusieron posibles sistemas productivos basados en blockchain y se analizó que ventajas y desventajas tendría implementar esta tecnología en este sistema.

El siguiente documento presenta en primera instancia el origen y contexto en que se desarrolla esta memoria: el proyecto base y el sistema de trazabilidad que propone para llevarlo a cabo.

POseriormente, se peresenta el problema, desarrollando un análisis de la solución inicial dando a conocer sus puntos débiles y como blockchain se convierte en una alternativa a analizar.

Luego, en las alternativas, se investigan las propiedades generales que tiene blockchain, sus tipos y algunas tecnologías específicas Ethereum y Hyperledger.

A partir de lo anterior se propone la realización de pruebas de concepto a fin de entender

la tecnología para luego analizar como sería una versión productiva del sistema comparada con la original concluyendo a partir de esto.

En resumen, esta memoria tiene como objetivo comprender si blockhain es una alternativa a este sistema, aprendiendo en el proceso las bases de esta tecnología y sus casos de uso más idóneos.

CAPÍTULO 1
MOTIVACIÓN Y CONTEXTO

1.1. MOTIVACIÓN PERSONAL

Desde el 2009 el mundo ha vivido una revolución silenciosa que promete grandes cambios para el futuro de muchas industrias. Blockchain ha sido desde su nacimiento con BitCoin una tecnología tan prometedora como mal entendida, incluso dentro de la propia comunidad informática, siendo reducida solo a criptomonedas.

Sin embargo, no es así. Blockchain es el nacimiento del internet del valor. Al día de hoy, el estándar para tranferir propiedad a través de internet es dependiente de una entidad central acreditadora que da fe las mismas, mientras que esta tecnología permite descentralizar esta tarea, delegándola a los miembros de la red.

Así surgen diversos usos, siendo el más conocido la transferencia de dinero sin intermediarios. Sin embargo, esta tecnología ha evolucionado permitiendo la creación de sistemas seguros y distribuidos aplicableas a organizaciones con estructuras complejas.

A modo de ejemplo, Estonia es el pionero en el uso de blockchain para el registro de interacciones en redes de trabajo gubernamentales, usando esta tecnología para el sistema de salud, de justicia y la policía siendo conocida como la sociedad digital mas avanzada del mundo. [1].

Por otro lado, consultoras tan grandes como Deloitte han hecho pública su apuesta por esta tecnología aplicada a cadena de suministros, asegurando que combinada con IoT resolverá problemas como fallas en efecto dominó, falta de visibilidad end-to-end y obsolescencia de tecnologías.[Deloitte, 2017]

En ese sentido, blockchain implica para mí una oportunidad para solucionar muchos problemas que nos afectan. Esta tecnología tiene el potencial de transparentar la contabilidad gubernamental, disminuir la burocracia en las organizaciones y atribuir responsabilidades, evitando la corrupción, aumentando la eficiencia y dando confianza a las personas.

Así, mi motivación es hacerme experto en esta tecnología y aprender a identificar sus casos de uso para de esta forma crear y ofrecer soluciones a problemas que nos afectan como sociedad.

[1] "Named 'the most advanced digital society in the world' by Wired". `https://e-estonia.com/`

1.2. CONTEXTUALIZACIÓN DEL PROBLEMA

1.2.1. INMOBILIARIA FUNDAMENTA

El proyecto se desarrolla en el marco de un desafío entregado por la inmobiliaria Fundamenta, el cual tiene como objetivo poner el valor el reciclaje para sus residentes y reducir el costo que conlleva realizar el mismo.

Actualmente esta inmobiliaria tiene como punto diferenciador hacer edificios sustentables acuñando el término de eco-inmobiliaria. Al entregar los edificios a la comunidad residente, estos se traspasan con contratos ya firmados con empresas que prestan los servicios del edificio, incluidos en estos los relacionados al reciclaje.

Sin embargo, el problema surge cuando la comunidad busca reducir los gastos comunes, siendo el reciclaje históricamente uno de los ítems más propuestos a ser eliminados de la plantilla de servicios por ser poco valorado.

Unido a lo anterior, los residentes reclaman que no tienen cómo comprobar si el proceso fue concretado. Las empresas recicladoras entregan informes, sin embargo no hay como verificar la veracidad de estos dado que el registro del proceso puede ser alterado por la misma empresa, dejando abierta la posibilidad de que el informe entregado no corresponda a la realidad.

Adicionalmente, la inmobiliaria no abarca todo el espectro de residuos reciclables, siendo este el caso de la basura orgánica, uno de los residuos más sencillos de tratar y más producidos por los hogares.

Cabe destacar que este problema no es solo de un edificio, sino que es generalizado a todos los edificios de Fundamenta y a modo general en todos los edificios que implementan reciclaje, ya que los mayores referentes en esta área, Triciclos [2], no implementan trazabilidad ni tratamiento de residuos orgánicos.

Así, por un lado se tiene a residentes disconformes que sienten que pagan un servicio de reciclaje que no saben si da frutos y por otro una inmobiliaria tratando de mantener su sello de eco-inmobiliaria una vez que el edificio es entregado.

1.2.2. SISTEMA CONCIENCITY

En ese contexto, el equipo Conciencity, formado en el marco de memorias multidisciplinarias UTFSM, propuso un sistema de reciclaje de residuos orgánicos autónomo dentro de sus edificios.

[2] Productos y servicios de Triciclos. `https://triciclos.net/productos-y-servicios/`

El prototipo inicial tiene como emplazamiento físico el piso 2 de 2 edificios, cada uno con 11 departamentos y consta del siguiente ciclo: recolección, transformación y almacenaje.

Durante la recolección el objetivo es individualizar los residuos de cada departamento (residente), asociando un identificador al peso del conjunto de residuos entregados. Estos residuos se entregan ya clasificados y son recolectados por un trabajador del edificio y posteriormente dispuestos en contenedores dentro del edificio.

La transformación inicia cuando un trabajador toma estos residuos y los pre-procesa para luego ser introducidos en la compostera. Esta cuenta con sensor de peso, para saber cuánto se está produciendo, y de humedad, para saber cuando el compost está listo para ser almacenado.

Finalmente, el almacenaje es la fase final donde el producto es sacado de la compostera y almacenado en un contenedor para ser vendido o usado dentro del edificio.

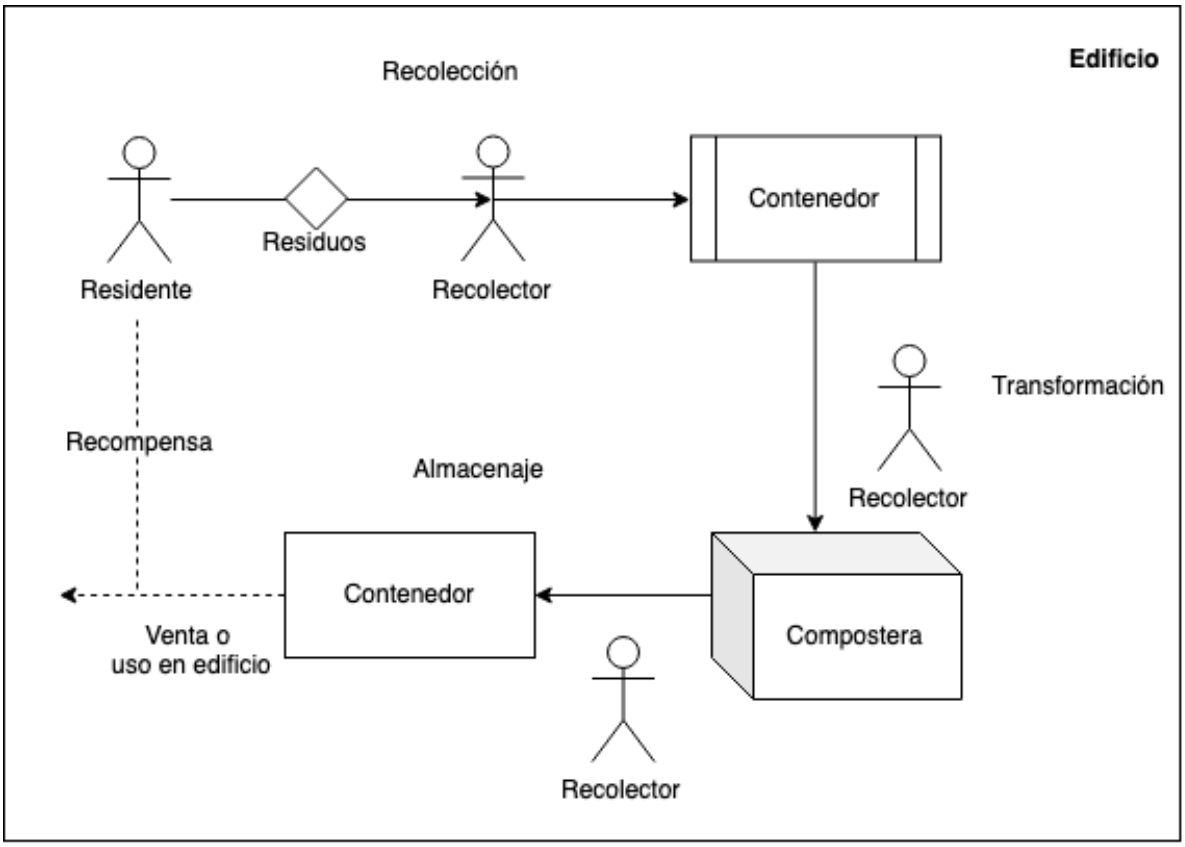


Figura 1: Sistema propuesto por el equipo Conciencity.
Fuente: Elaboración propia.

De ser vendido, los beneficios que son generados van al pago del gasto común del residente.

Por otro lado, si es usado dentro del mismo edificio, significa un ahorro en el ítem de compost para los jardines que tiene el edificio.

El sistema busca que durante todo el proceso los involucrados sepan cual es el estado en que se encuentra el residuo entregado por el residente, siendo necesario registrar todas las transacciones realizadas. De esta forma se aumenta la confianza en el proceso y adicionalmente los residentes pueden ser recompensados de forma proporcional a su aporte en la medida que el producto termina el ciclo.

La visión a futuro de Conciencity para este sistema, es que sea capaz de abarcar otros tipos de residuos, escalar a más nichos, como reciclaje en empresas e industrias, y acceder a otros modelos de negocio y formas de financiamiento, tales como certificados de carbono y bonos verdes.

En pos de lo anterior, el equipo tiene como desafío definir una arquitectura para el sistema que sea capaz de cumplir con las características de transparencia anteriormente descritas y que pueda escalar en el tiempo.

1.2.3. STAKEHOLDERS

A lo largo de este proceso, se ven involucrados diferentes entidades. La descripción de cada uno de ellos es la siguiente:

- Residente: Principal agente del sistema. Será quien aporte la materia prima mediante la separación en origen de los residuos, para posteriormente disponerlos en el sistema de recolección.
- Administrador: Es el encargado de mantener el registro de los residentes de la comunidad y además de los trabajadores de la misma.
- Recolector: Persona encargada de recolectar los residuos de la comunidad, realizar el preprocesado y sacar el producto terminado para su valorización y venta.
- Comité Administrador: Entidad encargada de velar por los intereses de la comunidad en torno a los servicios prestados por las demás entidades involucradas en el proceso de gestión de residuos. También propone, evalúa e implementa soluciones en favor de la comunidad.
- Inmobiliaria Fundamenta: Interesada en conocer estadísticas del proceso. Mirando a futuro, a esta le interesa saber como funciona el sistema en todos sus edificios, para de esta forma validarse como eco-inmobiliaria.
- Conciencity: Al igual que la inmobiliaria, busca conocer la efectividad del sistema. De esta forma busca aplicar mejora continua, brindar soporte y conocer el estado en todo momento.

- Entidades auditoras: revisan el proceso completo para poder acceder a nuevos negocios o modelos de financiamiento como bonos verdes y certificados de carbono.

1.2.4. PROPUESTA TECNOLÓGICA

Para implementar este sistema de residuos, el equipo Conciencity propuso un sistema basado las siguientes tecnologías:

Figura 2: Tecnologías propuestas por el equipo Conciencity.
Fuente: Conciencity.

El siguiente diagrama muestra la interacción que tendrían cada una de estas tecnologías

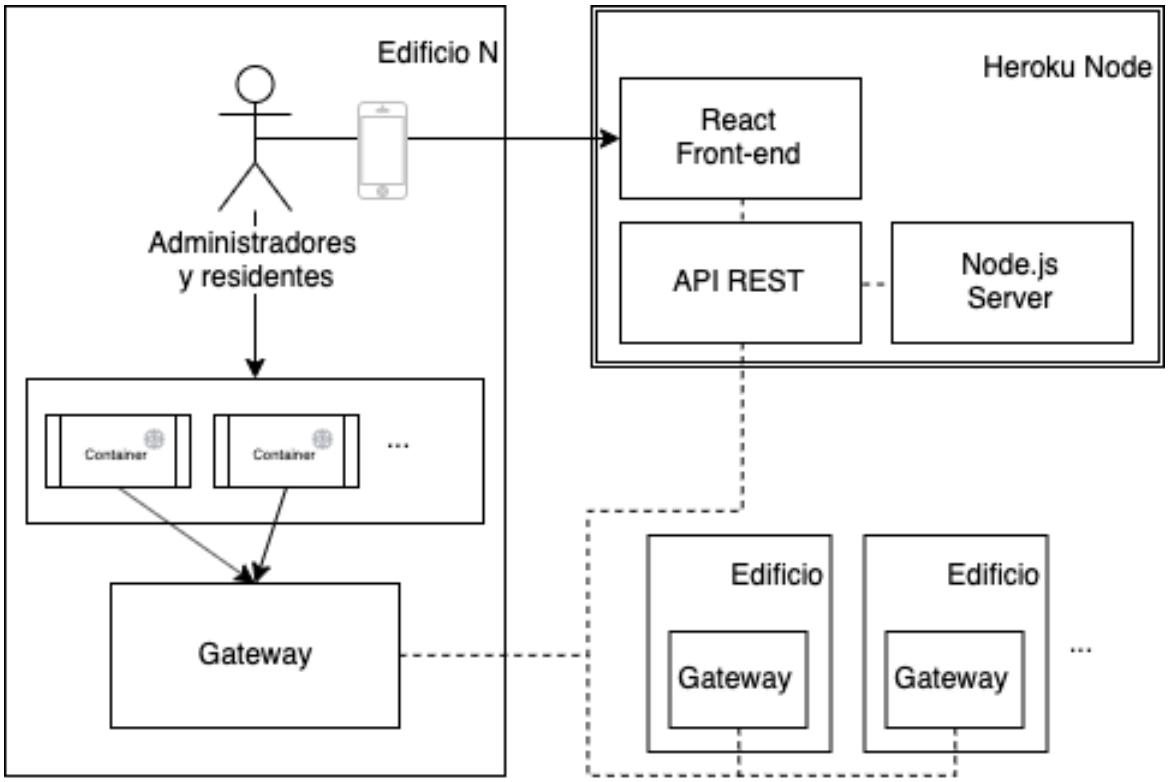


Figura 3: Diagrama de arquitectónico del sistema propuesto por Conciencity.
Fuente: Elaboración propia.

Cada edificio contaría con un gateway encargado de ser intermediario entre los contenedores y el servidor central. Estos nodos gateway pueden ser controlados remotamente desde el nodo central.

El servidor cuenta con un API que deriva las peticiones al back-end tanto desde los gateways como desde el front-end y las consultas externas.

Todos los usuarios, administradores y residentes, realizarían sus peticiones a través de una interfaz usuaria. Por otro lado las entidades enternas podrán hacer consultas vía API.

CAPÍTULO 2
DEFINICIÓN DEL PROBLEMA

En esta sección se analiza el sistema propuesto anteriormente a partir de sus fortalezas y falencias, dando énfasis en los posibles aspectos a mejorar del sistema.

2.1. ANÁLISIS DEL SISTEMA ORIGINAL

Dado el contexto y el sistema descrito, se desprenden ciertos objetivos base que se espera cumpla el sistema una vez implementado:

- Auditable: la existencia de *stakeholders* como la inmobiliaria o Conciencity hacen necesario un sistema capaz de entregar la historia del mismo, conociendo como se llegó al estado que se busca comprobar. Adicionalmente, se espera acceder a nuevos modelos de negocio y financiación en un futuro; certificados de carbono y bonos verdes, los cuales exigen ser auditados de forma periódica por entidades externas pudiendo necesitar una interfaz de consultas.
- Generar confianza: el sistema debe ser capaz de convencer a todos los *stakeholders* de que los procesos realizados y resultados obtenidos del sistema corresponden a lo que realmente fue procesado, algo que los sistemas actuales de reciclaje no cumplen.
- Disponble 24/7: el depósito de desperdicios ocurre a cualquier hora del día. Por otro lado el procesamiento y monitoreo de los mismos ocurre de forma preiódica, por lo que los sistemas deben estar siempre en funcionamiento.

2.1.1. ANÁLISIS DE CALIDAD DE SOFTWARE

El análisis de calidad de software previo al desarrollo es una forma de poder dilucidar si un sistema propuesto cumple con las expectativas esperadas, evitando así costos innecesarios en desarrollos que no entregan valor.

Para hacerlo, los sistemas se analizan usando diversos factores de calidad que ayudan a dilucidar el comportamiento de las plataformas en el futuro. En este caso, los factores son tomados como referencia del estándar dictado por el modelo McCall y se seleccionaron aquellos más influyentes en los requerimientos generales y en la arquitectura.

Aquellos mas relacionados al desarrollo del software, requerimientos funcionales en general, no serán tomados en cuenta en este análisis ya que esta es una fase de arquitectura y se asumirá que estos serán satisfactoriamente abordados.

Los factores a tomar en cuenta son los siguientes:

- Confiabilidad: Tiene que ver con la tolerancia a fallos. En este caso existe un punto crítico de falla asociado al nodo central donde se aloja tanto el front-end como el back-end. De haber una falla en ese punto todo el sistema queda inoperativo.

  Por el lado de los nodos, se cuenta con un gateway en cada edificio que puede fallar dejando al edificio sin registrar operaciones.

- Eficiencia: Tiene que ver con la velocidad de ejecución y almacenamiento. En este caso se puede concluir que cumple con su objetivo.

  En este sistema el foco está en el almacenamiento de datos debido al permanente monitoreo de los contenedores y entregas que realizan los residentes. El análisis de datos no va más allá de consultas simples a una base de datos. Por otro lado, la arquitectura escogida es un estandar de la industria con escalamiento vertical en la n siendo capaz de crecer en procesamiento de ser necesario.

- Integridad: Tiene que ver con la seguridad del sistema en general, tanto en aplicación como en almacenamiento y acceso a los datos.

  En este caso, a nivel de datos estos pueden ser modificados de forma unilateral por la empresa disminuyando la confianza en el sistema y manteniendo los mismos problemas de las plataformas que lo preceden.

  A nivel de aplicación, autorización y autenticación dependen del desarrollo de software y se asume que serán satisfactoriamente abordados.

- Mantenibilidad: Tiene que ver con los esfuerzos necesarios para localizar y reparar defectos del sistema. En este caso la arequitectura propuesta es un estandar y las tecnologías utilizadas tienen una amplia comunidad que las soporta, por lo que capacitar personas en su mantención y obtener herramientas de monitoreo no es una complicación.

- Interoperabilidad: Tiene que ver con la capacidad que tiene el sistema de comunicarse con otros. En este caso el sistema implementa una API, por lo que la comunicación con *stakeholders* está cubierta.

Dados los factores anteriores se puede saber si el sistema propuesto cumple con los requerimientos base.

La auditabilidad del sistema se ve más impactada por la interoperabilidad y la integridad de los datos. En esta arquitectura existe un sistema de consultas via API útil para comunicarse con las entidades auditadoras de ser solicitado. Por otro lado la comprobación de que los datos se derivaría solo del anásis de los datos. No existe un historia de transacciones contemplada que verifique como se llegó a los estados entregados por la plataforma.

Por otro lado, se puede decir que el sistema no mejora la confianza respecto a otros sistemas ya que la integridad de los datos sigue siendo dependiente a cabalidad de la empresa y por tanto son modificables.

Finalmente la disponibilidad 24/7 depende de la confiabilidad del sistema y la mantenibilidad del mismo. Debido al punto de falla único es posible tener caidas completas del sistema, haciendo que defectos de software y fallas en tiempo de ejecución impacten en gran medida en este item. Por otro lado, la mantenibilidad del sistema es buena teniendo en cuenta que el stack usado es estandar de la web en estos momentos y que todos los nodos pueden ser manejados de forma remota desde el nodo central.

Cabe destacar que al ser una arquitectura tan simple las posibilidades de mejorar y reparar estos problemas son grandes. La adopción de conceptos como escalamiento horizontal pueden reparar problemas de disponibilidad. La incorporación de registros de transacciones ayudarían a mejorar la auditabilidad, la confianza en el sistema e incluso la mantenibilidad del sistema.

Se puede decir que el sistma propuesto es suficiente para lograr los objetivos del sistema, sin embargo se buscan alternativas a esta arquitectura que puedan mejorar los aspectos mas débiles del sistema propuesto.

2.2. PROBLEMA

En el último tiempo blockchain aparece como una tecnología capaz de generar procesamiento y almacenamiento de datos de forma transparente, siendo estos inalterables y procesados con reglas públicas.

Además de su uso más conocido, las criptomonedas, se habla de un gran potencial en cadenas de valor, administración gubernamental, trazabilidad y traspaso de activos físicos.

La solución propuesta por Conciencity busca transparentar datos y dar confianza a clientes y auditores, por lo que blockchain suena como una alternativa a considerar.

Dado esto, el equipo busca saber si una arquitectura basada en esta tecnología ayudaría a mejorar los puntos débiles que tiene la arquitectura original propuesta y si es que es una mejor alternativa.

2.3. OBJETIVOS

2.3.1. OBJETIVO GENERAL

Dilucidar si blockchain es una mejor alternativa para registrar el proceso de reciclaje de residuos orgánicos en edificios propuesto por Conciencity.

2.3.2. OBJETIVOS ESPECÍFICOS

- Proponer una arquitectura basada en blockchain que cumpla con los requerimientos de la solución dada por Conciencity.
- Conocer los alcances, limitaciones y usos de esta tecnología.

CAPÍTULO 3
ALTERNATIVAS

Blockchain es el nombre que recibe una tecnología capaz de guardar registros de forma inalterable y distribuida mediante reglas conocidas por todos los nodos involucrados en un red de trabajo.

Existen diversas tecnologías basadas en este concepto y estas se diseñan para usos específicos, es decir en base a las necesidades que trata de resolver.

En esta sección se tratan las caracteristicas comunes que tienen las blockchain, los tipos y se detallan algunas tecnologías específicas a tomar en cuenta para el desarrollo de una propuesta de sistema productivo.

3.1. CARACTERÍSTICAS GENERALES

Blockchain es una base de datos distribuida que puede ser implementada de diversas formas. Para que esta funcione requiere de ciertos componenetes base necesarios: el ledger distribuido, los smart contracts y los algoritmos de consenso. A continuación se detallan estos componentes:

- Ledger distribuido: Es una estructura de datos y la base de blockchain. El ledger o libro de registros almacena de todas las transacciones ocurridas en la red de trabajo.

  Su estructura se compone de bloques de información correlativos ligados entre sí a través del uso de criptografía. Cada bloque contiene como información el hash del bloque anterior. Como resultado se obtiene que cualquier modificación de la cadena invalida la cadena completa.

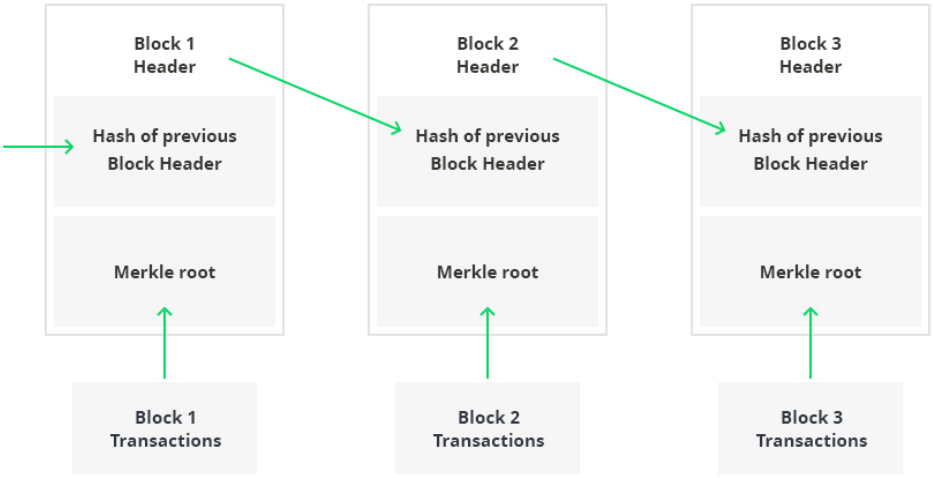


Figura 4: Estructura de datos básica de un ledger. Se observa como el hash del bloque anterior es parte del contenido del siguiete bloque.
Fuente: MLSDev.

Lo anterior implica que en la cadena solo se puede agregar información por lo que existe una traza completa de como se llego al estado actual de la cadena.

El ledger se describe como descentralizado debido a que es replicado en todos los participantes de la red, colaborando cada uno de ellos en su mantención.

Todo lo anterior le da a blockchain la capacidad de inmutabilidad, imposibilitando la modificación unilateral de los datos por cualquier participante y haciendo simple determinar la procedencia de la información que contiene.

- Smart contracts: Son la pieza que provee el control de acceso al ledger, logrando que a medida que es actualizado se mantenga consistente y que soporte las funciones comunes a una base de datos tradicional tales como realizar consultas y transacciones.

  Los smart contracts no solo son mecanismos claves para tratar la información y mantener de forma simple el ledger, sino que también permiten ejecutar de forma automática ciertos aspectos de las transacciones.

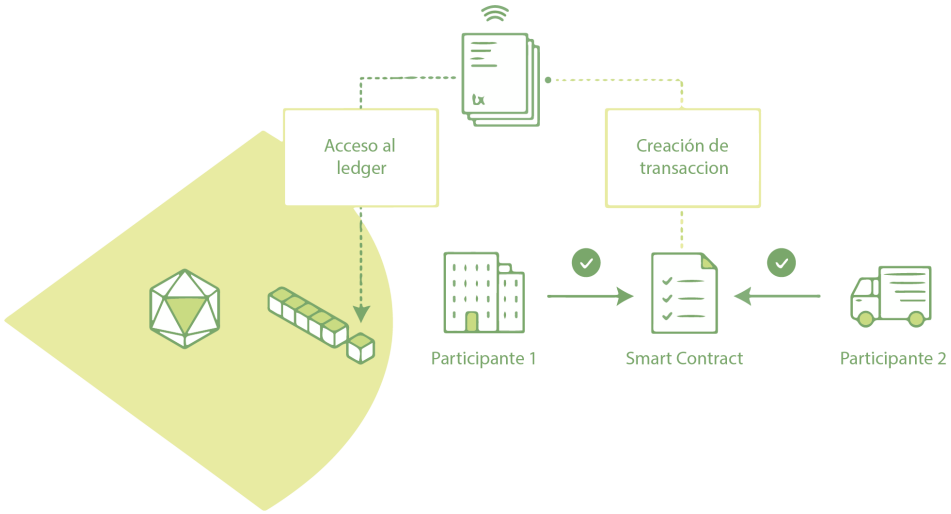


Figura 5: Representación de una transacción a través de un smart contract. Este autoriza y crea la transacción, dando acceso al ledger.
Fuente: Hyperledger.org

En ese sentido, un smart contract puede, por ejemplo, estipular el costo de enviar un objeto de forma automática dependiendo de la fecha en que llegó al contenedor, su peso y la fecha en que llegó a destino. Estos términos son aceptados por ambas partes y son almacenados en el ledger, permitiendo que los cobros se hagan automáticamente por la misma red una vez que el objeto es recibido.

- Consenso: Es el mecanismo para mantener las transacciones del ledger sincronizadas en todos los pares. Este es un algoritmo que asegura que los registros sean actualizados solo cuando las transacciones sean aprobadas por los participantes específicos de la red que tienen ese rol. Una vez aprobadas, se encarga de actualizar a todos los pares con el mismo ledger en el mismo orden.

  El algoritmo de consenso es dependiente del tipo de red que lo necesita. Dependiendo de esto, los algoritmos pueden ser comparados en su velocidad, escalabilidad y finalización. A continuación se presenta una tabla comparativa con el fin de ilustrar este comportamiento.[5]

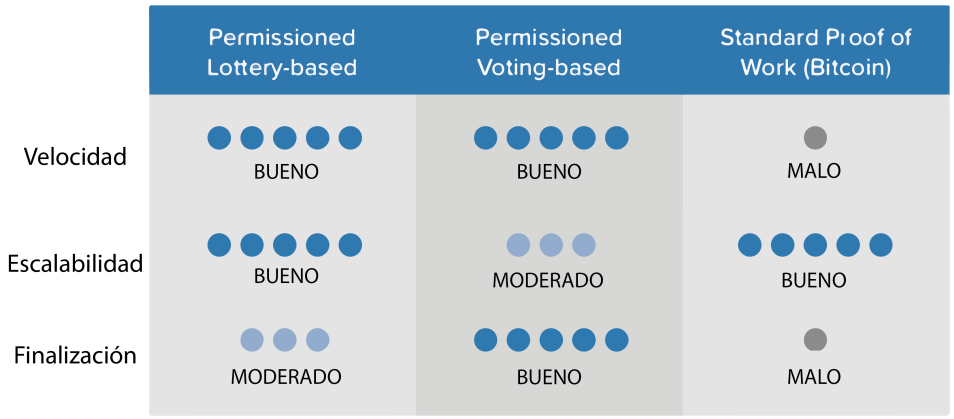


Figura 6: Comparación de los tres tipos generales de algoritmos de consenso. Fuente: Hyperledger Architecture, Volume 1. [Hyperledger, 2018]

La combinación de estos tres componentes hacen que blockchain sea considerada como una base de datos distribuida capaz de mantener información de forma transparente e inalterable, sin embargo todas estas características dependen del contexto y tipo de red, algo que se verá en capítulos siguientes.

Estas características hacen que ya se hayan implementado sistemas financieros, de propiedad intelectual, juegos y cadenas de valor basadas en esta tecnología.

3.2. TIPOS DE BLOCKCHAIN

Las blockchaina a modo general se clasifican por una combinación de dos criterios: los permisos de envío y lectura de transacciones, y los permisos de procesamiento de transacciones.

Cada uno de estos da pie a ventajas y desventajas que impactactan en la integridad del ledger, su seguridad, y la velocidad de procesamiento de las transacciones.

El uso de cada tipo depende del contexto en que se busque utilizar esta tecnología siendo de vital importancia la elección del tipo de blockchain para lograr los objetivos que se buscan.

- Públicas - Sin permisos: redes donde cualquiera puede unirse para realizar lecturas, escrituras y procesar transacciones.

  En general estas redes son más grandes alcanzando los cientos de miles de ususarios. Se asume desconocimiento entre miembros de la red y por lo tanto existe desconfianza entre pares.

  En general tienen velocidades de transacción lentas ligadas a los protocolos de consenso que deben implementar para evitar fraudes. Sus datos son prácticamente inmodi-

ficables por la cantidad de nodos que los resguardan y la censura no puede ocurrir debido a que todos los nodos pueden verificar transacciones.

Son las mas conocidas y en ellas se construyen Dapp's, aplicaciones distribuidas donde las reglas de negocio están regidas en su gran mayoría por smart contracts y que requieren de wallets para su uso.

- Públicas - Con permisos: redes donde cualquiera puede unirse para realizar lecturas y escrituras, pero que el procesamiento de transacciones es restringido a ciertos pares. Lo anterior ayuda a mejorar la velocidad de transacción haciendólas aptas para sistemas de alta demanda.

  Este sistema goza de una alta transaccionalidad al ser nodos de confianza los que procesan información, sin emabargo están propensas a la censura por parde de los que procesan las transacciones. La modificación de los datos es prácticamente inmodificable pues el ledger puede ser mantendo por nodos públicos.

  En general estas redes se usan para entidades privadas que prestan servicios a públicos y que necesitan ser transparentados. En general, este tipo es usado por grandes consorcios de financieras para dar servicios de transferencia internacional.

- Privadas - Sin permisos: redes donde una entidad decide quienes pueden unirse para realizar lecturas, escrituras y procesar transacciones.En otras palabras, una vez permitido el acceso tiene permisos totales.

  Su uso está poco extendido, existen pocas implementaciones con este paradigma y su comunidad es muy reducida. En ellos cada smart-contract es una cadena en si misma y para participar de ella se debe solicitar a pares ser ingresado.

- Privadas - Con permisos: redes donde una entidad decide quienes pueden unirse para realizar lecturas, escrituras y un subconjunto de ellos para procesar transacciones. En otras palabras, solo algunos procesan transacciones.

  Este es un paradigma más corporativo y su uso mas extendido es la trazabilidad de activos entre compañias. Ayuda a homologar la información compartida entre instituciones y sus velocidad de transacción es rápida, ya que los protocolos de consenso utilizados son mas laxos al ser una red de pares de confianza.

  La posibilidad de censura y modificación del ledger es muchísimo mayor que en las redes anteriores ya que la información y el procesamiento reside en pocos nodos.

3.3. TECNOLOGÍAS

Existe una gran cantidad de tecnológias disponibles con el potencial de implementar la plataforma ideada por Conciencity. Las siguientes son sólo algunas de las alternativas, existiendo muchas más opciones.

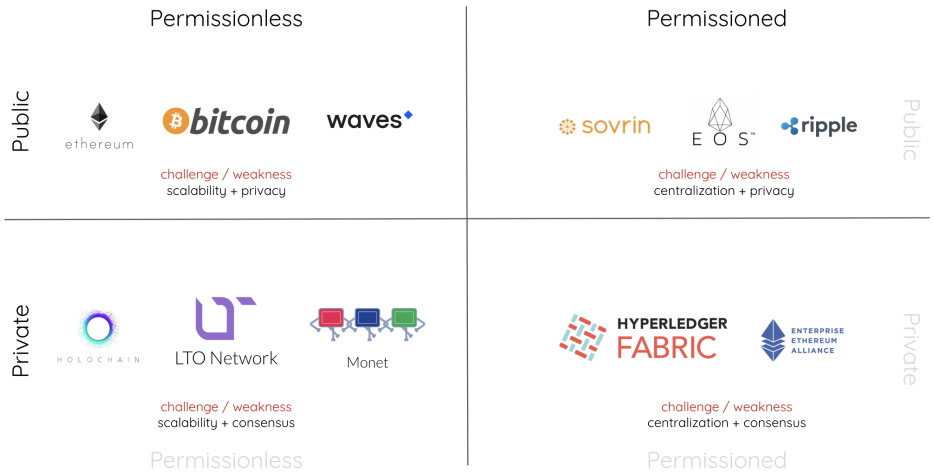


Figura 7: Implementaciones de blockchains según su clasificación.
Fuente: Medium [Arnold, 2018]

La elección de cual usar es difícil debido a que muchas de estas tecnologías estan en fases de desarrollo siendo inestables y con poca comunidad.

Dentro de ellas, existen dos tecnologías que se han convertido en referentes de dos paradigmas radicalmente diferentes: Ethereum (pública - sin permisos) y Hyperledger Fabric (privada - con permisos). En ambos casos existen plataformas que actualmente se encuentran en producción y una comunidad creciente de desarrolladores que las mantienen.

Por todo lo anterior, son las candidatas a ser implementadas en el sistema y a continuación se detallan sus características.

3.3.1. ETHEREUM

Ethereum se define como una plataforma global, descentralizada y de código abierto para aplicaciones descentralizadas. Fuel lanzada el 2015 siendo de las primeras capaces de soportar smart contracts de forma nativa.

El ledger es la única fuente de datos del sistema. El bloque es guardado en una base de datos llave-valor encriptada en hashes.

Los smart contracts soportan el estándar de tokens ERC20, el cual es utilizado para representar activos de forma única y haciéndolos transferibles. Por otro lado tiene su propio lenguaje de programación, Solidity, con una amplia comunidad de desarrolladores.

Esta blockchain puede ser implementada de dos formas: como pública - sin permisos y como privada - con permisos. Dependiendo de el tipo de red que busquemos implementar el

protocolo de consenso funciona de forma diferente.

- Pública - Sin permisos: utiliza el protocolo de consenso Proof of Work, teniendo el Ether como su moneda, la cual cumple con las funciones de incentivar el código eficiente y el minado.

  El protocolo funciona resolviendo un desafio criptográfico de alto procesamiento. Todos los mineros, nodos de procesmiento, de la red compiten por resolverlo. Una vez realizado el resultado es comprobado por el resto de mineros. El bloque es agregado a la cadena y se envia a todos los nodos de la red. El minero que lo resuelve recibe ether como recompensa a su trabajo.

  Su foco está en la creación de Dapps, aplicaciones que guardan sus datos y comprueban su integridad de en la blockchain mediante contratos inteligentes. También es utilizado como publicador de datos.

  Sus ventajas y desventajas asociadas al sistema que se busca construir son las siguientes:

- Privada - Con permisos: utiliza el protocolo Proof of authority. Con este protocolo el concepto de minado no existe y por lo mismo no tiene una moneda asociada.

  En el ciertos nodos son escogidos por la red para procesar las transacciones. Verificadas las transacciones por todos los nodos, el bloque es añadido a la cadena y enviado a todos los pares evitando la fase de alto procesamiento.

  Su foco está en organizaciones que buscan compartir datos con reglas claras y estandarizando los mismos, evitando problemas de incompatibilidad de datos entre compañia y reduciendo los costos de mantención de las bases de datos.

3.3.2. HYPERLEDGER FABRIC

The Linux Foundation creó Hyperledger en el año 2015 para atacar el nicho industrial de blockchain. En vez de declarar un único estándar, esta institución utilizó un enfoque colaborativo en desarrollo de la plataforma junto a la comunidad.[6]

Hyperledger Fabric es uno de los proyectos asociados a blockchain dentro de Hyperledger. Como otras plataformas de este estilo, esta se compone de un ledger, el uso de smart contracts y un sistema que permite a los participantes manejar sus transacciones mediante un algoritmo de consenso. La diferencia de Hyperledger Fabric está en que es una red privada y que requiere permisos para que usuarios la usen. En otras palabras, esta blockchain no permite identidades anónimas y para participar en esta red se debe pasar por un Membership Service Provider (MSP).

Hyperledger Fabric ofrece diversas opciones ajustables a la red. El ledger puede ser almacenado en múltiples formatos, los mecanismos de consenso pueden ser cambiados y diferentes MSP pueden son soportados.

Por otro lado, ofrece la capacidad de crear canales, permitiendo que grupos de participantes de una red tengan ledgers adicionales y ocultos del resto, útil en entornos de negocio donde se hace concesiones comerciales a ciertos participantes y a otros no.

A continuación se verá en profundidad los detalles de cada componente de este sistema.

- Ledger compartido: El ledger de esta plataforma tiene dos componentes principales: el world state y el transaction log. Cada participante tiene una copia de del ledger perteneciente a cada canal en que participa.

  El world state describe el estado de del ledger en un momento dado. Es la base de datos del ledger. El transaction log guarda todas las transacciones que han influido en el world state; es su historial de actualizaciones.

  El ledger tiene distintas formas para almacenar sus datos, pero defecto este usa LevelDB, una base de datos clave-valor. Por otro lado, el transaction log solo guarda los valores actualizados en cada transacción, por lo que su estructura no es modificable.

- Smart contracts: En Hyperledger Fabric, los smart contracts están escritos en chaincode y son invocados por aplicaciones externas solo cuando esta necesita interactuar con el ledger. En la mayoría de los casos, chaincode interactúa solo con el componente world state y no con el transaction log, debido a que la mayoría de las peticiones son solicitudes de información y no escrituras.

  El chaincode puede ser implementado en diversos lenguajes de programación. Actualmente tiene soporte para Go, Java y Javascript, pero se tiene pensado la implementación para otros lenguajes.

- Consenso: Hyperledger Fabric permite escoger el algoritmo de consenso dependiendo de las relaciones existentes entre los miembros de la red. Como ocurre con la privacidad de ciertas organizaciones, hay un amplio espectro de necesidades dependiendo de las relaciones entre los miembros de la red; desde estructuras altamente jerarquizadas hasta muy horizontales.

  De esto dependerá el algoritmo a escoger. En el caso de Hyperledger Fabric, existen actualmente dos algoritmos: Kafka, que no soporta usuarios maliciosos, y SBTF, que si lo hace, pero requiere más tiempo para alcanzar un consenso. A pesar de lo anterior las velocidades de transacción han marcado hasta 20000 transacciones por minuto.

  El consenso se compone de tres partes: endoso, ordeno y comprometo. Cuando un cliente propone una transacción se envía a los pares o nodos endosantes que simulan la transacción, si se cumplen las reglas preestablecidas en los contratos inteligentes (chaincode) entonces los nodos endosantes le devuelven al cliente una confirmación firmada y validada por ellos.

  La aplicación del cliente devuelve el endoso validado al servicio de pedidos (ordering service) que es el encargado de ordenar todas las transacciones en un bloque; el servicio de pedidos es acoplable y permite ordenar las transacciones utilizando diferentes

mecanismos de consenso. Hyperledger Fabric actualmente incluye Solo (únicamente para testear la red) y Kafka (producción). Una vez que el servicio de pedidos ha ordenado las transacciones endosadas en un bloque se las entrega a los nodos comprometidos, estos nodos verifican los endosos y validan los resultados de las transacciones antes de ser comprometidas a la cadena de bloques (blockchain).

Si la transacción falla, es decir, si el par comprometido determina que los datos no coinciden con el estado global actual, la transacción ordenada en un bloque seguirá incluyéndose en ese bloque, pero se marcará como no válida, y el estado global no se actualizará. Los nodos comprometidos son responsables de agregar bloques de transacciones al libro contable compartido y actualizar el estado mundial. Pueden tener contratos inteligentes, pero no es un requisito. Por último, los nodos comprometidos notifican a la aplicación cliente el éxito o el fracaso de la transacción.

Canales que permiten la visibilidad de las transacciones solo para las partes interesadas.

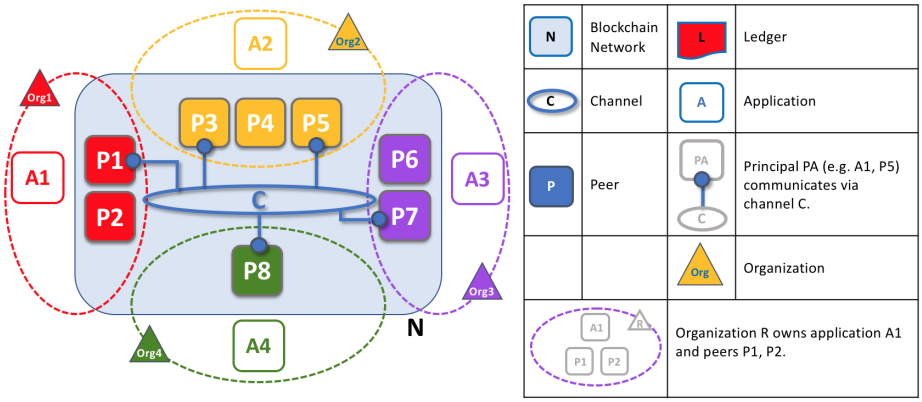


Figura 8: Diagrama de interacciones entre nodos para el consenso en Hyperledger Fabric. Fuente: Hyperledger.org.

CAPÍTULO 4
PROPUESTA DE SOLUCIÓN

En este capítulo se realiza un primer análisis de las tecnologías blockchain aplicadas al sistema Conciencity y se propine un marco de acción para verificar si aportan al sistema.

4.1. ELECCIÓN DE TECNOLOGÍAS

Para dilucidar si blockchain es una alternativa para el sistema Conciencity se propuso realizar pruebas de concepto (PoC) con el problema simplificado, una de una red privada con permisos y otra con una red pública sin permisos.

La elección de las tecnologías se hizo teniendo en cuenta las ventajas y desventajas de cada una de las tecnologías investigadas. La comparación se puede ver en la siguiente tabla.

Tabla 1: Comparativa de ventajas y desventajas .
Fuente: Elaboración Propia.

	Ventajas	Desventajas
Ethereum Público - Sin permisos	- Stakeholders pueden ver datos sin necesidad de interfaces. - Red configurada. - Inmutabilidad de datos asegurada. - No puede haber censura.	- Costo por transacción y deploy de contratos. - Protocolo de consenso lento.
Ethereum Privado - Con permisos	- Protocolo de consenso rápido. - No hay costo por transacción ni deploy.	- Posibilidad de censura. - Inmutabilidad de datos depende de la confianza en la red. - Se debe configurar la red. - Requiere de una interfaz para compartir datos o que el stakeholder mantenga un nodo.
Hyperledger Fabric	- Protocolo de consenso rápido. - Flexible para escalamiento o nuevas funcionalidades.	- Posibilidad de censura. - Requiere de definir roles, arquitectar y configurar la red. - Requiere el dominio de muchas tecnologías. - Curva de aprendizaje lenta. Inmutabilidad de datos depende de la confianza en la red.

Desde el punto de vista de prueba de paradigma, Ethereum destaca como la mejor alternativa ya que ofrece dos protocolos que probar a partir del mismo código, los dos más utilizados en la industria.

Hyperledger requiere de una arquitectura y roles complejos, algo muy específico de esa tecnología y que no contribuye al objetivo general que es saber si blokchain como paradigma es una alternativa.

4.2. PROPUESTA

La propuesta final consiste en hacer dos PoC en Ethereum, implementando los dos paradigmas que pueden ser desarrollados con el, los dos más utilizados dentro dentro del mundo blockchain.

Con esto se buscó comprender en profundidad y de forma empírica las ventajas y limitaciones de blockchain en general, más allá de cualquier paradigma.

A partir de esta PoC, se busca proponer posibles arquitecturas productivas que resuleven el problema y analizar si estas mejoran los factores de calidad del capítulo 2.

CAPÍTULO 5
IMPLEMENTACIÓN

En esta sección se muestra la adaptación del problema a una PoC y como esta se matrializó en la tecnología Ethereum.

5.1. MODELAMIENTO

Se realizó un modelamiento que se abstrae del contexto inicial residuos.

Un usuario hace una entrega, la cual genera un item inicial en un contenedor perteneciente a una etapa. El item puede ser traspasado a distintos contenedores y etapas de forma fraccionada. En una última etapa los items se transforman en un producto. Los contenedores guardan el peso e items que contiene. Cada acción, entrega, traspaso y transformación, tiene responsables.

El detalle de las tres acciones del flujo es el siguiente:

- Entrega: Cuando un usuario realiza una entrega y como consecuencia crea el item inicial en un contenedor.
- Traspaso: Cuando items, o fracción de ellos, pasan a otro contenedor.
- Transformación: Cuando items, o fracción de ellos, pasan a conformar un producto.

Estas acciones permiten conocoer cierta información básica acerca del estado del sistema:

- Conocer el estado de una entrega.
- Conocer el estado de un contenedor.
- Conocer el estado de una etapa.
- La historia de un producto.
- Conocer los responsables de un contenedor.

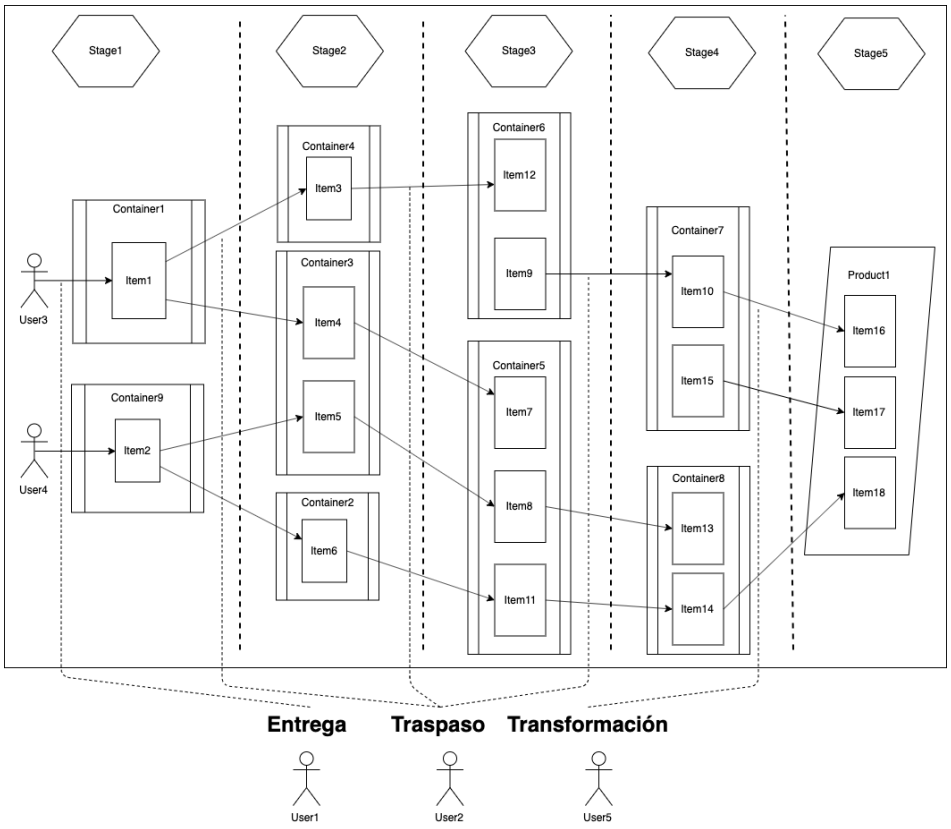


Figura 9: Diagrama del modelado realizado para la prueba de concepto. En el se aprecian todas entidades e interacciones posibles entre ellas.
Fuente: Elaboración propia.

El testing en ambas implementaciones fue el mismo y consistió en un caso base del flujo de creación de dos productos y posterior consulta de datos. Todos los aspectos respectivos al rendimiento se compararon en base a ellos. Las interacciones medidas se dividieron en transacciones y consultas:

Transacciones: cambian el estado de la blockchain y se propagan por toda la red. Lo más costoso en tiempo, computación y dinero. En este caso corresponden a las entregas, traspasos y transformaciones.

Consultas: Se hacen a nivel de nodo local y no cambian el estado del ledger. En este caso corresponden a conocer el estado de una etapa, de una entrega, de un contenedor, la historia de un producto y los ultimos responsables de un contenedor.

Todas las interacciones fueron medidas obteniendo la cantidad de gas y el tiempo necesario para que cada una de ellas se completara.

En el siguiente diagrama se muestra las transacciones realizadas en el flujo de testing y los valores finales de cada contenedor.

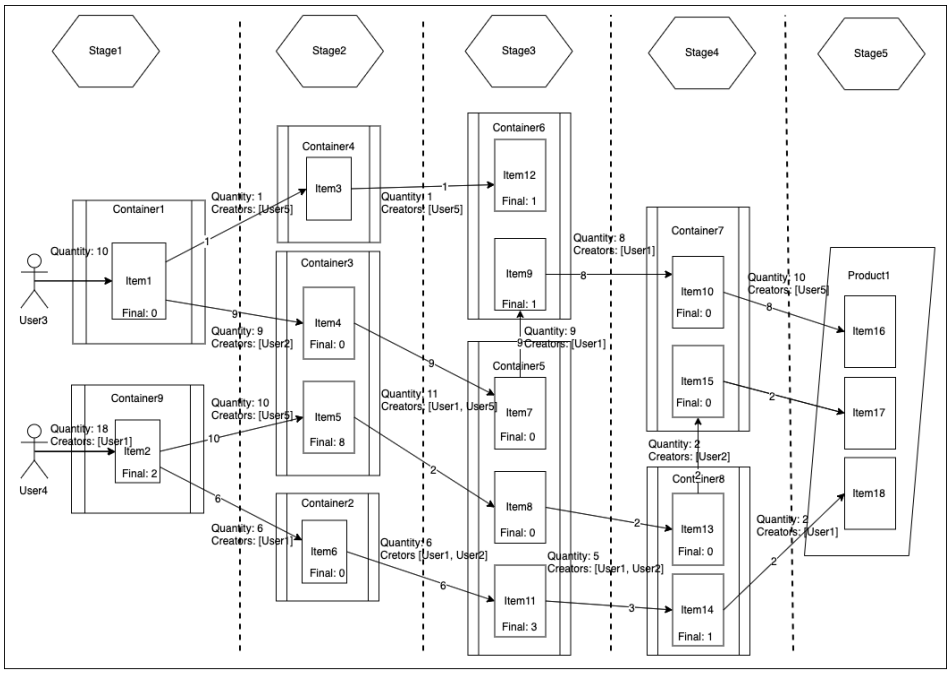


Figura 10: Flujo de transacciones aplicadas durante la prueba de concepto.
Fuente: Elaboración propia.

5.2. IMPLEMENTACIÓN

Se implementó un programa en Node.js encargado de realizar las pruebas, simulando tanto la entrada de datos por sensores como la interacción con el usuario a través de interfaz gráfica.

El stack utilizado fue el siguiente:

- Geth: Nodo de Ethereum implementado en Go.
- Node.js: Runtime de javascript. Simula entradas de sensores e interacciones de usuario.
- Ganache: Simulador de red Ethereum para pruebas locales.
- Truffle: Framework de desarrollo Solidity.
- Solidity: Lenguaje para la blockchain de ethereum.

El uso de el estandar ERC20 para representar los paquetes fue descartado debido a que en este sistema los objetos pueden ser divididos y reunificados de forma permanente. Este estandar no es una buena opción para tratar con fracciones de objetos, por lo que simplemente se llevará registro de las cantidades.

La cantidad de nodos, la administración de las wallets, la cantidad de ether y su forma de obtención dependen de la implementación, pública o privada, y se detallan en cada sección a continuación:

5.2.1. ETHEREUM PÚBLICO

Se propone usar una testnet de ethereum, Ropsten, para realizar las pruebas públicas debido a que la obtención de ether es gratis en esta red. El tamaño de esta red al momento de las pruebas es de 50GB y debe ser descargada para poder hacer transacciones y consultas.

Estas pruebas se hacen en un nodo local conectado a la red, el cual maneja todas las wallets. Estas fueron cargadas de ether usando un faucet, fuente de ether para pruebas.

El flujo de ejecución de las pruebas fue el siguiente:

1. Se levanta el nodo y se une a la red Ropsten.
2. Se hace deploy desde truffle, el cual lo sube a la red Ropsten.
3. Se ejecutan las pruebas desde Node.js

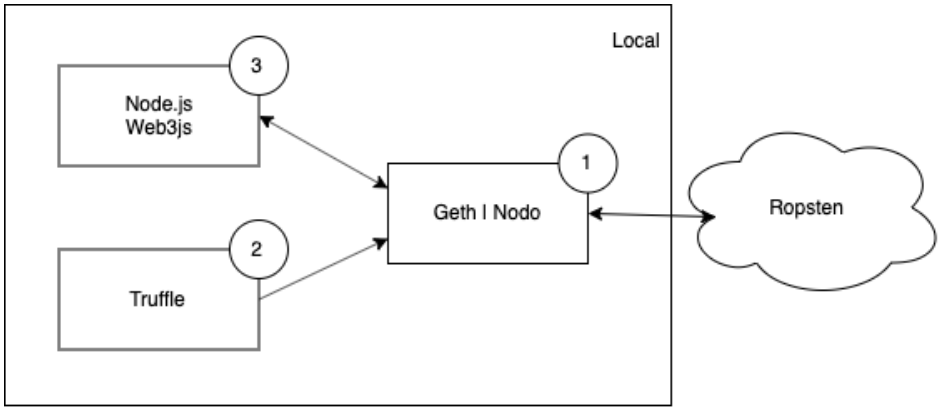


Figura 11: Flujo de ejecución de las pruebas públicas.
Fuente: Elaboración propia.

5.2.2. ETHEREUM PRIVADO

Se levantó un cluster de nodos privados de ethereum, simulando tres nodos: una de la empresa y dos de edificios.

Para poder hacerlo se debe configurar una red desde el inicio. Esto implica generar un bloque genesis, el primero de la cadena. Su configuración genera 1 bloque por segundo, implicando que anualmente 13GB de datos e generan como cota mínima.

El nodo1 maneja dos cuentas, la del dueño del contrato y la de un administrador. Además funciona de bootnode, aquel nodo fijo con el cual el resto de nodos encuentra a sus pares.

El nodo2 maneja 6 cuentas, 1 administrador y 5 usuarios. A este se conecta el programa de pruebas Node.js.

El nodo3 se creó para hacer la prueba de adición de nodos a la red. Maneja una cuenta y es usada para procesar transacciónes provenientes de los otros dos nodos.

El flujo de ejecución de las pruebas fue el siguiente:

1. Se levanta el nodo1.
2. Se levanta el nodo2. Automáticamente se conecta con el nodo1.
3. Se hace deploy del contrato.
4. Se levanta el nodo3 conectándose con el nodo1 y luego con el nodo2
5. El nodo 3 es autorizado por los otros dos nodos para ejecutar y guardar transacciones.
6. Se hacen las pruebas.

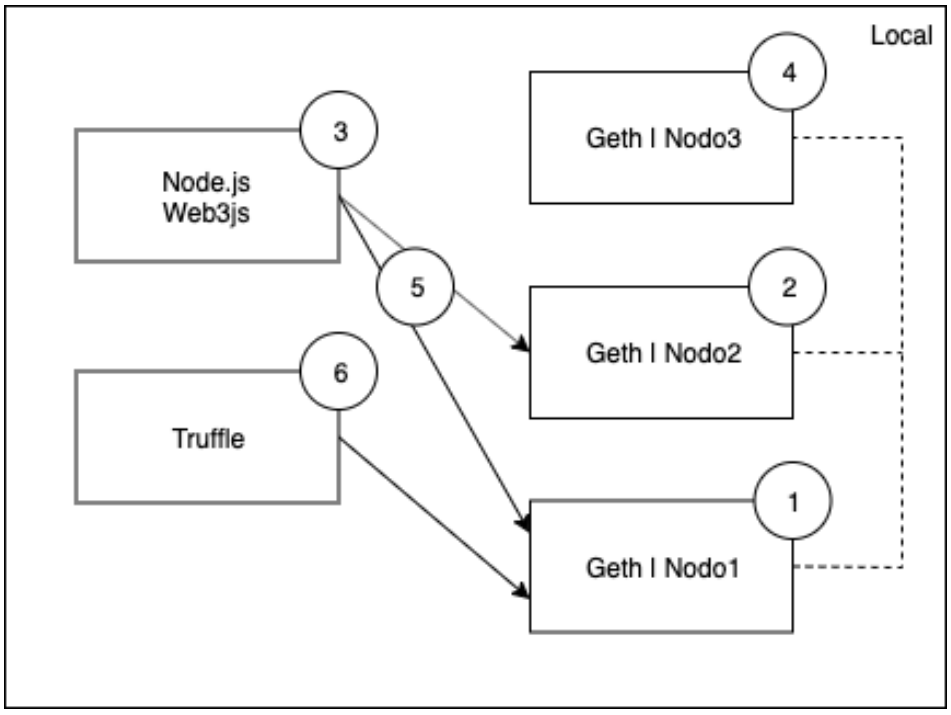


Figura 12: Flujo de ejecución de las pruebas privadas.
Fuente: Elaboración propia.

5.3. RESULTADOS

Se observó que los tiempos de ejecución fueron muy diferentes dependiendo del protocolo de consenso utilizado. Lo anterior calza con lo esperado y se puede observar en la tabla 2

Por otro lado, luego de la implementación se observó que el costo en gas no varía dependiendo del protocolo. La razón de esto es que el gas es una medida de costo costo computacional y por lo tanto depende de el algoritmo ejecutado y del estado del ledger. Al ser las mismas pruebas, el gas usado es el mismo.

Además de lo anterior, el tiempo no solo es mayor en una red pública, sino que es más inestable, concordando con lo esperado. La saturación de la red tiene mucha influencia en la ejecución de las transacciones del sistema y esto se puede observar en la figura 13

Tabla 2: Resultado de tiempo y gas asociado a las pruebas de concepto.
Fuente: Elaboración propia.

	Ethereum público - sin permisos	Ethereum privado - con permisos
Tiempo mínimo por transacción[ms]	2015	1011
Tiempo promedio por transacción[ms]	25277	2006
Tiempo máximo por transacción[ms]	106370	2094
Tiempo total PoC[ms]	5434483	436834
Gas mínimo	289046	
Gas promedio	329047	
Gas máximo	931530	
Gas Total	70745042	

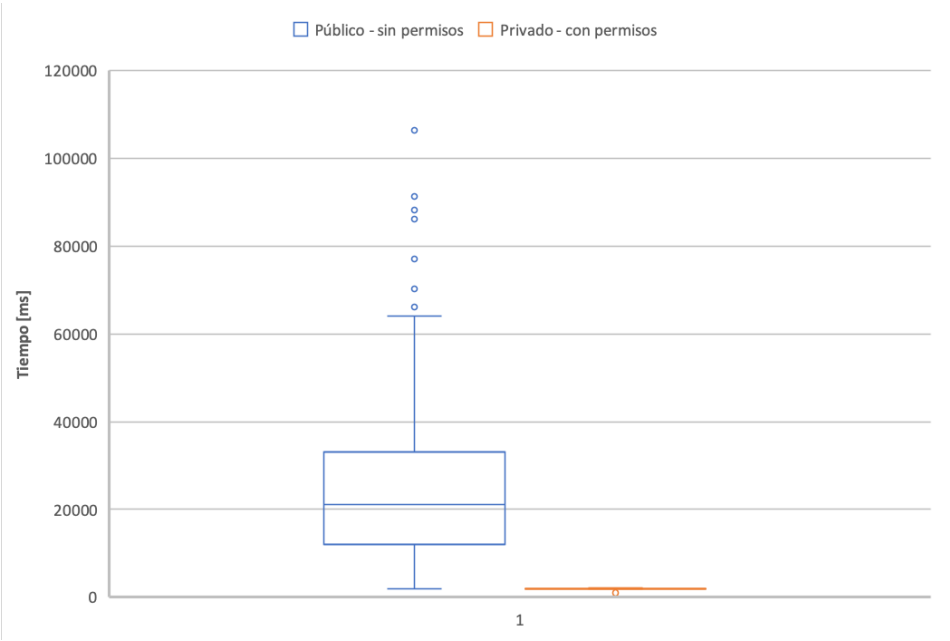


Figura 13: Boxplot asociado a la velocidad de transacción en cada tipo de red.
Fuente: Elaboración propia.

CAPÍTULO 6
VALIDACIÓN DE LA SOLUCIÓN

En esta sección se proponen dos posibles arquitecturas productivas basadas en el conocimiento aprendido con las pruebas de concepto, siendo luego son analizadas y comparadas con los factores de calidad el sistema original.

6.1. ARQUITECTURAS PROPUESTAS

Dado lo aprendido en las pruebas de concepto se propusieron dos posibles arquitecturas basadas Ethereum: una pública - sin permisos y un privada - con permisos.

6.1.1. PROPUESTA DE RED PÚBLICA - SIN PERMISOS.

En esta opción cada edificio tendrá un nodo conectado a la mainnet de ethereum. Las interacciones de los usuarios se realiza a través de una interfaz usuaria que consulta a un nodo central.

- Nodo central: está expuesto a internet y tiene la función de recibir consultas provenientes de la interfaz usuaria de la que harán uso los usuarios de los edificios; residentes y recicladores.

  De no existir, cada usuario tendría que llevar un nodo en su dispositivo, algo inviable por la cantidad de procesamiento y almacenamiento necesario.

- Nodo de edificio: se conectan directamente a los contenedores y a la mainnet de Ethereum donde se envían las transacciones para ser confirmadas. La posibilidad de que estos minen se descarta porque se requiere de gastos en hardware y electricidad elevados y baja probabilidad de minado de un bloque.

  Las transacciones de los contenedores son registradas por estos nodos. Esta decisión se toma debido a que así mejora la resistencia a fallos del sistema.

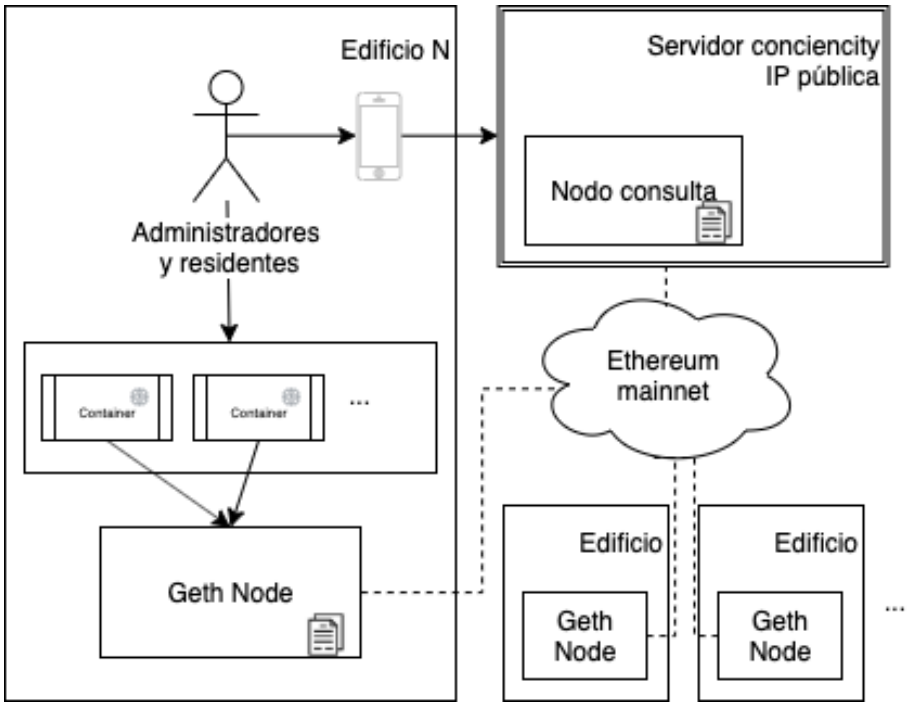


Figura 14: Diagrama de arquitectura del sistema basado en Ethereum público - con permisos propuesto para Conciencity.

Fuente: Elaboración propia.

6.1.2. PROPUESTA DE RED PRIVADA - CON PERMISOS.

En esta arquitectura la red, al igual que en la red pública, está compuesta por un nodo central expuesto a internet y un nodo por edificio.

La principal diferencia es el protocolo que utilizan estos nodos para guardar información y donde se procesan las transacciones.

- Nodo central: está expuesto a internet y tiene la función procesar transacciones y de recibir consultas provenientes de la interfaz usuaria de la que harán uso los usuarios de los edificios; residentes y recicladores.

  Además de lo anterior, cumple la función de bootnode, es decir, este es un punto de consulta donde el resto de nodos de la red puede consultar la direccion de sus pares y publicar la propia.

  De no existir, cada usuario tendría que llevar un nodo en su dispositivo, algo inviable por la cantidad de procesamiento y almacenamiento necesario. Por otro lado, cada nodo debería ser configurado de forma manual cada vez que se agrega un nuevo nodo a la red.

- Nodo de edificio: se conectan directamente a los contenedores y a al resto de la red. Se propone que cada uno de estos nodos procese transacciones de tal forma que estas sean verificadas por muchas entidades.

  Estos nodos deben tener una copia del ledger para poder procesar las transacciones.

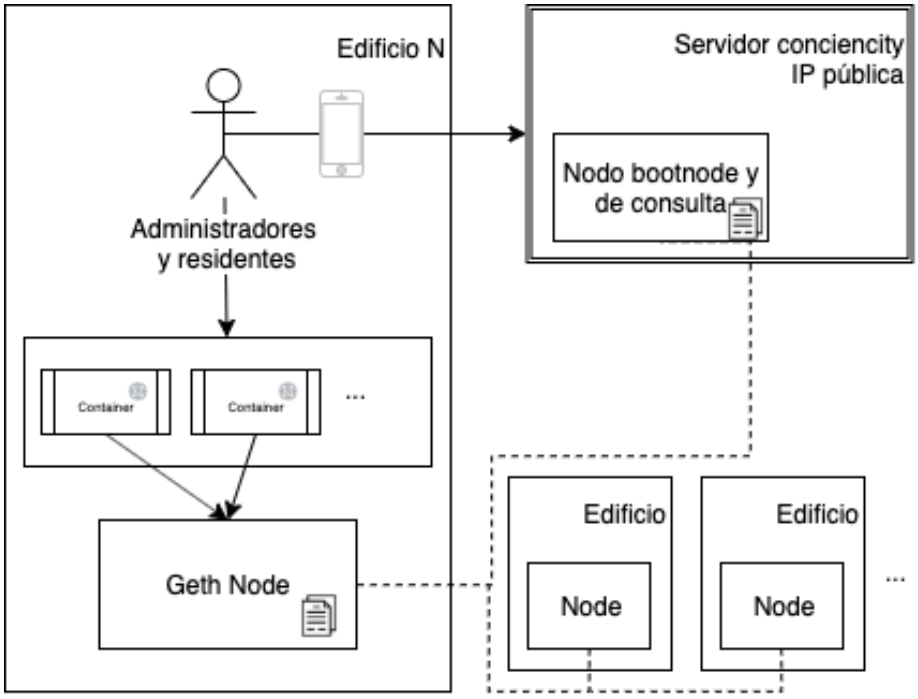


Figura 15: Diagrama de arquitectura del sistema basado en Ethereum privado - sin permisos propuesto para Conciencity.

Fuente: Elaboración propia.

6.2. ANÁLISIS DE ARQUITECTURAS

6.2.1. FACTORES DE CALIDAD

- Confiabilidad: Mejora respecto a la arquitectura original, pero depende de ciertos factores y del protocolo utilizado.

  A nivel de registro de entregas, traspasos y transformaciones, una falla en un nodo impide el funcionamiento del sistema solo para ese edificio, manteniendo su funcionalidad toda la red.

  A nivel de consulta de usuario, al hacerse en un nodo central es posible que exista una falla global, sin embargo no crítica por tratarse de consultas de residentes y recicladores. Esto podría subsanarse con servicios externos como Infura, una API para interactuar con contratos inteligente, suponiendo un costo extra.

En términos genrales, la infraestructura y los protocolos ultilizados hacen que una falla a nivel general sea muy dificil dependiendo de algunos requisitos en cada protocolo:

- Red pública: La posibilidad de que la transacción no sea procesada existe en la medida que se envíe poco gas en cada transacción disminuyendo la confianza en el sistema. En sistemas con transacciones mas manuales es sencillo de notar un problema, pero cuando se trata de nodos remotos enviando transacciones de forma periódica el problema es mayor.

  En caso de ocurrir, existen procesos para reenviar transacciones con más gas, aumentando los costos y teniendo que desarrollar un sistema de monitoreo de transacciones para poder subsanar esta problemática.
- Red privada: De estar caído el nodo central la inserción de un nuevo nodo no puede realizarse disminuyendo la confianza en el sistema. Sin embargo, no es una falla crítica y es sencilla de monitorear. Simplemente es un factor a considerear.

Por todo lo anterior y tomando en cuenta los supuestos, considero que este sistema basado en blockchain es más tolerante a fallos y resiliente, pues la caída y recuperación un nodo es imperceptible para el resto de nodos reduciendo el riego de fallo crítico considerablemente.

- Eficiencia: En general empeora. En ambas implementaciones es necesario que muchos nodos ejecuten el código para que la transacción se confirme en la blockchain. Sin embargo existen diferencias en cada implementación:

  - Red pública: Como se observa en la tabla 2 y en la figura 13, los tiempos de ejecución de transacción son muy inestables y lentos comparado con la red privada.

    Esto concuerda con la inestabilidad conocida de la mainnet, donde ofreciendo el gas promedio a los mineros, la transacción puede tardar entre 15 segundos y 5 minutos. Lo anterior depende de congestión de la red, un factor incontrolable y solo monitorizable.

    Esto genera un problema grave de incertidumbre en el el sistema. En momentos de red saturada es muy posible que la red quede inoperativa sin almacenar depósitos ni estados de contenedores a menos que se monitorize la inserción de los datos y reenvíen las transacciones con más gas, sin embargo nada asegura un tiempo estable de transacción.
  - Red privada: Con este protocolo los tiempos de procesamiento son estables comparado con la red pública. Comparádo con la solución original los tiempos de latencia son presumiblemente mayores por el protocolo de consenso, alcanzando un promedio de 2 segundos por transacción, razonable y util para el caso de uso.

    Aquí no existe el concepto de mineros, sino que de firmantes. Al no tener que generar hashes de forma exhaustiva el proceso en cada nodo sólo es ejecutar y agregar a la cadena siendo un proceso más lento que una inserción en una base de datos, pero aceptablemente rápido.

Por otro lado, observando la figura 13, la estabilidad es muy buena, siendo en comparación con la red privada un sistema más confiable en tiempos de respuesta.

- Integridad: Mejora o se mantiene dependiendo del protocolo.

  - Red pública: Mejora considerablemente. Con este protocolo la mantención del ledger y ejecución de las transacciones dependen de entidades externas a la empresa. Dado lo anterior, la modificación de los datos es imposible por cualquier entidad y la censura de transacciones no puede ocurrir.
  - Red privada: Se mantiene. Existe un ataque que consiste en que si una persona logra controlar la mayoría de los nodos, esta puede cambiar e incluso reemplazar la blockchain completa. Cabe destacar que este problema es independiente del protocolo.

    En este caso, Conciencity es el administrador de los nodos por lo que su capacidad de modificar el ledger es absoluta. El problema se resolvería si cada administración de cada edificio fuera dueña de los nodos, algo que en la práctica es dificil que suceda.

    Las comunidades buscan que los servicios que contratan funcionen sin grandes mantenciones y menos con la necesidad de mantener gente con conociminetos técnicos avanzados dedicados a esto.

    Desde el punto de vista de la censura, ocurre exactamente lo mismo. Conciencity al ser dueño de los nodos podría vetar o modificar transacciones provenientes de ciertos nodos, violanto la integridad del sistema.

- Mantenibilidad: Empeora respecto al sistema original. Durante la prueba de concepto salieron a la luz muchas sutilezas en el proceso de desarrollo y operación que hacen difícil mantener, monitorear y escalar un sistema en esta tecnología.

  Algunos desafíos y problemas encontrados durante el proceso de implementación fueron que una vez publicados los contratos no se pueden cambiar, que el costo de procesamiento y publicación es alto en la red pública, el debugging requiere de un alto grado de conocimiento de la Ethereum Virtual Machine, que la ejecución de transacciones puenden tardarse mucho o no ocurrir por falta de gas y que la sincronización con la red de pruebas requiere de mucho tiempo y recursos de hardware para ser completada. Dado lo anterior se detectan las siguientes consecuencias:

  - Se deben adoptar patrones de desarrollo específicos para contratos inteligentes que permitan crear contratos modulares y modificables. A modo de ejemplo, una actualización de contrato no es posible y debe ser manejado a través de contratos con variables a que apuntan a contratos actualizados.
  - El código debe ser muy eficiente aprendiendo y usando todas las sutilezas del lenguaje para mantener transacciones baratas: variables con tipo de dato ajustado al tamaño del dato, evitar ciclos, confirmar precondiciones, entre otros.

- El proceso de testing es fundamental ya que cada deploy y procesamiento de transacción tiene un costo asociado en ether. La recomendación, en especial en la red pública, es quitar toda la lógica posible del contrato y centrarse solo validaciones para disminuir el costo de la transacción, utilizando back-end tradicional para el procesamiento de los datos.
- El largo del contrato influye en el costo de ejecución. Cabe la posibilidad que el contrato quede guardado en dos bloques, aumentando el costo por transacción.
- Se debe tener un sistema de monitoreo de transacciones para el sistema público en caso de que las transacciones no se estén ejecutando. Ademas este sistema debería guardar las transacciones y reenviarlas con mas gas de forma automática.
- Una caída en un nodo implica mucho tiempo de baja para un edificio en particular. La sincronización de los nodos tienen una fase final llamada ”state trie”muy dependiente del hardware que tenga el nodo y responsable de que el nodo de pruebas Ropsten no pudiera ser levantado.

Blockchain vuelve todo aspecto más complejo y requiere de aprender muchos patrones nuevos si se quiere hacer un sistema completo basado en esta tecnología.

La cantidad de desarrolladores en este lenguaje, Solidity, es muy poca y su curva de aprendizaje es lenta si se quiere hacer un sistema mantenible en el tiempo. Comparándolo con Node.js, Mongo y todas las tecnologías usadas por el sistema original, este sistema tiene bastante desventaja.

- Interoperabilidad: Mejora levemente o empeora dependiendo del protocolo.

  - Red pública: En este caso existe una leve mejora debido a que los datos son públicos y una entidad auditora puede acceder a ellos sin necesidad de ninguna interfaz otorgada por Concienity.

    A pesar de lo anterior, pensar que un sistema productivo puede funcionar de esta froma es inocente. Una entidad auditora tendría que replicar todas las transacciones de un contrato para conocer el historial del mismo, para luego llegar a comprobar el estado final de la blockchain.
  - Red privada: Empeora. Debe crearse una API donde las entidades externas puedan consultar además de reconstruir el historial de transacciones de no desarrollarse el contrato. De ser este el caso, se mantiene igual que en el sistema original.

    En un futuro, si esta tecnología se vuelve estándar sería posible que esta entidad mantuviera un nodo con la red privada para monitorear, pero por el momento no se considera una alternativa.

6.2.2. OTROS FACTORES RELEVANTES

- Costos: En el sistema original los costos estaban asociados directamente al funcionamiento del hardware. En el caso de blockchain, estos costos se mantienen en el caso

privado, sin embargo no ocurre así en el caso público.

Los costos para la red pública aumentan debido a la cantidad de gas que se está dispuesto a pagar para que un deploy de contrato o una transacción tenga prioridad y como se vio en muchos factores de arriba, este es fluctuante e incontrolable por parte del sistema. Esto viene dado por como funciona el protocolo proof of work: Los mineros buscan ganar ether por su costo de procesamiento.

El stack de transacciones de cada minero en general está dado por la cantidad de gas ofrecida por aquel que envía la transacción, por lo que una transacción que ofrece poco gas es enviada al final del stack demorándose mucho en ser ejecutada o incluso no ocurriendo nunca.

Para evidenciar el problema, a continuación se presenta una tabla que muestra como varían los costos en base a la cantidad de ether ofrecida por gas (gas price). Esta cantidad cambia las posibilidades de que una transacción sea procesada por los mineros.

Tabla 3: Costos asociados a la prueba de concepto en dólares y a la posibilidad de realización de la transacción.

Fuente: Elaboración propia.

Posibilidad de ser escogido en las siguientes 200 transacciones	50.94 %	86.79 %	98.11 %
Costo por transacción de 329046 gas	$0.05	$0.46	$0.57
Costo total de los test realizados 70745042 gas	$12.44	$99.60	$124.51

El resultado de esto es la posibilidad de que los costos por transacción fluctúen hasta en un orden del 1000 %.

- Explotación de datos: Si bien todo el historial de transacciones esta guardado en la blockchain, los datos históricos no son facilmente obtenibles. La explotación de datos se vuelve compleja si se busca acceder recurrentemente a, por ejemplo, los cambios que ha tenido un contenedor.

  Existen dos formas de tener data histórica, al menos en ethereum: Programando un contrato que guarde en un array valores históricos o replicando todas las transacciones realizadas en un contrato.

  La primera opción implica aumentar el costo por transacción y la segunda implica un lento proceso que, primero, recorre toda la blockchain en búsqueda de transacciones de ese contrato. Segundo, ejecuta todas las transacciones guardando cada uno de los resultados.

Existe una opción a esto que implica infraestructura adicional: guardar en una base de datos externa el histórico y usar blockchain solo como una verificación, sin embargo esto aumenta los costos y complejidad del sistema. Es por esto que cobra sentido la estructura de Hyperledger Fabric, que mantiene un transaction log para este tipo de casos.

6.2.3. IMPACTO EN LOS OBJETIVOS BASE

A continuación se aplican los factores analizados anteriormente a los objetivos base definidos para este sistema en la seccion 2.1.

- Auditable: El sistema en blockchain si lo logra mejorar. A pesar de ser engorroso obtener el historial de transacciones y los valores asociados en un momento determinado, la información siempre estará y puede ser reconstruida.

  La obtención de la información en bruto es más sencilla para las entidades auditoras en el caso público ya que estas se encuentran en la cadena. En el caso de la red privada es igual que en el caso del sistema original, necesitando una API para consultar.

  Cualquier modificación de datos será evidente a menos que se modifique toda la cadena, algo posible en la opción privada, pero no en la red pública. En el peor de los casos, es exactamente igual que con el sistema original.

- Generar confianza: En el caso público mejora muchísimo en el aspecto de veracidad de los datos, pero esto se ve mermado por los tiempos y costos intestables de transacción además de la posibilidad de que estas incluso no se lleguen a realizar.

  En el caso privado no existen mejoras. La posibilidad de que Conciencity cambie toda la cadena de bloques hace que toda la infraestructura pierda sentido, siendo mejor el sistema original al ser mas simple de implementar.

- Disponible 24/7: Se logra, pero no es una mejoría sustancial debido al paradigma blockchain, sino que al protocolo específico utilizado.

  En el caso del sistema público, esta es más dificil de lograr no por una falla de hardware, sino por lo inestable que es el sistema de envío de transacciones en el protocolo proof-of-work.

  Por el lado del sistema privado si se observa una mejoría enorme. Dado que cada nodo es independiente del resto en la medida que la mayoría funcione, una falla en uno de los nodos no implica una caída del sistema completo.

  Desde el punto de vista de la mantención, esta impacta de forma muy negativa. El equipo necesario para mantener un sistema como este es difícil de encontrar y de entrenar por ser un paradigma muy diferente, haciendo complejo reparar bugs, monitorear, diagnosticar y generar nuevas funcionalidades de forma continua y segura.

CAPÍTULO 7
CONCLUSIONES

A lo largo de este documento se estudió a fondo la pertinencia de blockchain en un sistema de reciclaje de residuos orgánicos que está en una fase muy temprana de desarrollo, investigando si esta tecnología mejora un sistema opropuesto.

Respondiendo a la pregunta principal: ¿Es blockchain una mejor alternativa para este sistema? La respuesta es no, al menos no con las tecnologías investigadas. Muchas razones técnicas se detallan a lo largo del documento que refuerzan esto, pero las principales tienen mucha más relación con los paradigmas generales que con los detalles de implementación.

Las razones que descartan una red pública son principalmente la alta incetidumbre en costos y tiempos de transacción. Un sistema altamente automatizado no se puede permitir una fluctuación latente del orden del 1000 % en sus costos o del 6000 % en tiempo de ejecución, más pensando en un eventual escalado a muchos nodos. Esto es un problema asociado a la mayoría de las blockchains públicas - sin permisos.

Respecto a una red privada, la principal razón para descartarla es que pierde la principal característica de blockchain, su integridad. Esto hace que el sistema se convierta una compleja máquina que no mejora la posibilidad de modificación unilateral de datos, siendo mucho más ventajoso el sistema original por su simplicidad. Esto hubiese sido cierto incluso usando un sistema basado en Hyeprledger Fabric. En otras palabras, el paradigma de red privada - con permisos es incompatible con un sistema administrado por una sola entidad.

Que una misma tecnología implementada con distintos paradigmas difiera tanto en sus características deja evidencia que blockchain más que una tecnología concreta, es un paradigma que debe entenderse a alto nivel para dilucidar en que casos es pertinete utilizarlo. En este documento se investigaron solo dos tecnologías, Ethereum y Hyperledger Fabric, de las cuales se implementó solo una de ellas obteniendo diferencias notables basadas solo en el paradigma.

Dado lo anterior cabe hacerse una pregunta ¿En qué casos blockchain es un aporte?

A nivel de paradigma, una red pública sin permisos es ideal para plataformas de representación y transferencia de objetos. Plataformas de publicación de derechos de autor, autenticación de activos públicos con cadenas de valor aplicadas a vinos o representación activos digitales únicos como personajes de videojuegos. Todos los sistema anteriores buscan declarar la propiedad y origen de cosas, pero que no requieren de un alto grado transaccional.

Por otro lado, las redes privadas tienen su fuerte cuando distintas entidades colaboran y buscan tener un sistema unificado con reglas claras. Alianzas entre empresas que buscan compartir información son ideales para esta tecnología pues permiten estandarizar la re-

cepción, permisos y formato de los datos. El interés por resguardar sus intereses hacen que cada una de los involucrados sean administradores de los nodos, imposibilitando la modificación unilateral de los datos y manteniendo todas las propiedades de blockcahin.

Algunos ejemplos concretos de blockchain aplicado de forma exitosa se asocian a instituciones tan importantes como el gobierno de Estonia, donde el sistema de salud, la policía y al justicia tiene esta tecnología aplicada para hacer un estado mas eficiente y transparente, dando a entender que el potencial de esta tecnología es enorme, solo hay que saber donde aplicarlo.

Las notarías son perfectas para esta tecnología si se arma una red de notarios e instituciones públicas cada una con su nodo. Estas entidades, en teoría defienden sus propios intereses manteniendo la integridad del sistema y haciendo una infraestructura descentralizada y transaparente. Los costos son adecuados para la firma de documentos y mucho menores que los que se cobran hoy en estas instituciones.

En ese sentido esta investigación ha sido muy esclarecedora. Ha permitido comprender cuales son las principales carácteristicas de cada tipo de blockchain dando herramientas para dilucidar cuáles son los mejores casos para aplicar esta tecnología, comprendiendo que cada paradigma y cada tecnología tiene sus propias particularidades.

A modo personal, considero que una investigación más a fondo de este caso de uso con otros paradigmas, otra tecnología o mas infraestrucutra es innecesario.

El mejor legado y camino natural para avanzar a partir de este documento es aplicar blockchain en un sistema idóneo para el, que se beneficie de esta tecnología. En otras palabras, es una invitación a explorar casos en que esta tecnología brille y permita hacer instituciones más transparentes, eficientes, menos burocráticas y en consecuencia mejorar la vida de las personas a través de ella.

REFERENCIAS BIBLIOGRÁFICAS

[Arnold, 2018] Arnold, D. (2018). The rise of private permissionless blockchains — part 1. https://medium.com/ltonetwork/the-rise-of-private-permissionless-blockchains-part-1-4c39bea2e2be.

[Deloitte, 2017] Deloitte (2017). Continuous interconnected supply chain using blockchain internet-of-things in supply chain traceability. https://www2.deloitte.com/content/dam/Deloitte/lu/Documents/technology/lu-blockchain-internet-things-supply-chain-traceability.pdf.

[Hyperledger, 2018] Hyperledger (2018). Hyperledger architecture, volume 1. https://www.hyperledger.org/wp-content/uploads/2017/08/Hyperledger_Arch_WG_Paper_1_Consensus.pdf.