Gestión Segura de Certificados Digitales de Eventos Científicos mediante Blockchain *

María J. Peregrina-Pérez[1] and Juan Boubeta-Puig[2]

[1] Escuela Superior de Ingeniería, Universidad de Cádiz, Cádiz, España
mariajesus.peregrinaperez@alum.uca.es
[2] Grupo UCASE de Ingeniería del Software, Departamento de Ingeniería Informática, Universidad de Cádiz, Cádiz, España
juan.boubeta@uca.es

Resumen Uno de los grandes desafíos de la digitalización es la generación de identidades digitales seguras. Actualmente los sistemas que gestionan estas identidades siguen siendo mayoritariamente centralizados, como es el caso de los que acreditan la asistencia a eventos científicos. Para afrontar este problema, este artículo propone gestionar de forma segura la emisión, la entrega y el almacenamiento de certificados digitales de eventos científicos mediante la tecnología blockchain. A esta tecnología se unirá *InterPlanetary File System* (IPFS), que permitirá almacenar los certificados de asistencia a eventos científicos de forma descentralizada, mientras que blockchain gestionará los metadatos de dichos certificados. De esta forma, la acreditación pasará a ser descentralizada, agilizando este proceso de forma segura, trazable, inmutable y transparente. Los resultados obtenidos demuestran la eficacia de esta solución, la cual integra tecnologías novedosas y que permite gestionar certificados de manera segura y descentralizada a usuarios y empleados de una universidad.

Keywords: Blockchain · IPFS · Certificado digital

1. Introducción

La acreditación e identidad digital es uno de los grandes desafíos actuales de la digitalización [4]. Esta información debe ser segura, inmutable, transparente y trazable, para así poder verificar la veracidad de dicha información. En los últimos años han aparecido varias tecnologías para afrontar este reto, entre ellas la tecnología *blockchain*.

La tecnología *blockchain* consiste en un libro mayor distribuido que se estructura en una lista vinculada de bloques. Dichos bloques contienen, a su vez, un conjunto ordenado de transacciones. Se utilizan *hashes* criptográficos para asegurar el enlace de un bloque a su predecesor [19]. Una función *hash* genera identificadores únicos, de tamaño fijo, a partir de una información dada.

* Trabajo financiado por MCIN/AEI/10.13039/501100011033/ y FEDER (proyecto AwESOMe PID2021-122215NB-C33), y por el Plan Propio UCA 2022-2023.

Una desventaja conocida de la *blockchain* es su alto coste de almacenamiento de datos. Para afrontar dicho problema, una red blockchain puede ser integrada con IPFS, un sistema distribuido para almacenar y acceder a archivos, de forma que en la blockchain solo se almacenen los metadatos de los ficheros previamente almacenados en una IPFS.

Partiendo de estas tecnologías, se permite suplir una de las necesidades que tienen actualmente el Personal Docente e Investigador (PDI), y el estudiantado de las universidades y, en general, cualquier persona: acreditar ante los organismos que así lo requieran la asistencia a un evento científico a través de un certificado que sea seguro, inmutable, transparente y trazable.

El resto del artículo se estructura como sigue. En la Sección 2 se detallan los fundamentos de ambas tecnologías. En la Sección 3 se presenta la propuesta de este trabajo. En la Sección 4 se muestran los resultados obtenidos. En la Sección 5 se describe una serie de trabajos relacionados. Finalmente, en la Sección 6, se presentan las conclusiones y las líneas de trabajo futuro.

2. Fundamentos

En esta sección se describen las tecnologías *blockchain* e IPFS.

2.1. Blockchain

Un sistema basado en *blockchain* consiste en una red *peer-to-peer* de máquinas, llamadas nodos. Contiene una estructura de datos para la réplica del libro mayor en la red [13]. Dicha red se compone de un protocolo de red que define derechos, responsabilidades, y procesos como la verificación y validación entre los nodos. Además, establece cuáles son los mecanismos para agregar nuevos bloques, así como la autorización y autenticación de las transacciones [19]. Dependiendo de su accesibilidad, la *blockchain* se puede dividir en dos tipos: públicas o privadas (también conocido como permisionadas). En las públicas los usuarios pueden unirse por sí mismos cuando lo deseen, mientras que en las permisionadas un usuario central (normalmente un administrador) deberá autorizar el acceso a la red.

Según la naturaleza de la propuesta presentada en este artículo, se ha decidido trabajar con redes permisionadas. De esta forma las diferentes organizaciones serán las que decidirán qué usuarios pueden participar en la red, así como definir sus políticas de seguridad. Esto permite un mayor control y privacidad de los datos tratados, algo esencial en organizaciones de tipo gubernamental o educativo (universidades). Por ello, se ha seleccionado Hyperledger Fabric [14] como plataforma para la red *blockchain*. Sus puntos diferenciadores frente a otras plataformas son su compatibilidad con diferentes protocolos de consenso, su permisionado de redes, sus contratos inteligentes pueden ser creados con lenguajes de programación de uso general (Java, Go y Node.js) y su privacidad y confidencialidad tanto de transacciones como de los contratos inteligentes [2]. Además,

cuenta con un nodo llamado *orderer* (también conocido como "nodo de ordenación") que hace esta ordenación de transacciones, que junto con otros nodos de ese tipo, forman un servicio de ordenación, garantizando así que cualquier bloque validado por el par sea definitivo y correcto. Los libros mayores distribuidos no pueden bifurcarse como lo hacen en muchas otra redes *blockchain* públicas [11].

Un concepto intrínseco a las *blockchain* es los *smart contracts*. Los contratos inteligentes son simplemente programas almacenados en una *blockchain* que se ejecutan cuando se cumplen condiciones predeterminadas [6,16]. Por lo general, se utilizan para automatizar la ejecución de un acuerdo para que todos los participantes puedan estar seguros de inmediato del resultado, sin la participación de ningún intermediario ni pérdida de tiempo. También pueden automatizar un flujo de trabajo, activando la siguiente acción cuando se cumplan las condiciones [1].

2.2. Interplanetary File System

IPFS es un sistema distribuido para almacenar y acceder a archivos, sitios web, aplicaciones y datos [5].

Dado que se trata de un sistema descentralizado, IPFS debe tener un control de dónde se encuentra el contenido que será solicitado por los usuarios. Para ello, IPFS utiliza el direccionamiento de contenido, a diferencia de las URLs tradicionales que se sustentan en el direccionamiento por localización (dónde se encuentra dicho contenido).

IPFS genera un *hash* criptográfico a partir del contenido del elemento, que será el identificador de éste. Por tanto, en caso de que se modifique el contenido del fichero, su identificador también cambiará. De esta manera, se permite verificar la integridad del contenido solicitado por el usuario, a diferencia del direccionamiento por localización.

Hay dos tipos de redes IPFS: públicas o privadas. Por defecto la red IPFS es pública, por lo que cualquier persona puede participar en ella y ver su contenido. El único requisito es poseer un nodo IPFS.

Por otro lado, en las redes privadas además de un nodo IPFS, se requiere de una clave para participar. Esta clave se llama *swarm key* y es generada por el nodo principal. Cuando se desea que alguien se una a la red, el poseedor de la clave pasará por un canal seguro una copia al usuario. Con esta privacidad se consigue que los contenidos no sean accesibles por cualquier persona mediante el *Content IDentifier* (CID). Debido a esta privacidad, se ha decidido desplegar una red IPFS privada.

3. Propuesta

El objetivo de nuestra propuesta es crear un sistema que proporcione seguridad para la gestión de ficheros de forma descentralizada.

El sistema final permitirá a los usuarios crear o eliminar los ficheros subidos a la red, y cada acción quedará registrada en la *blockchain* de forma automática.

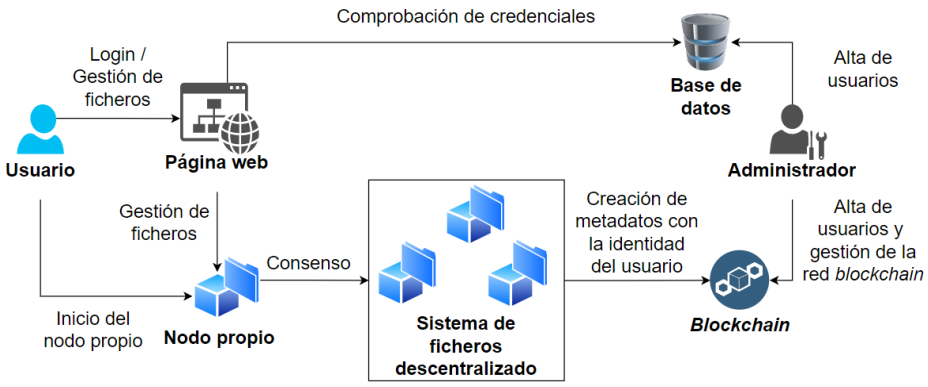

Figura 1. Arquitectura del sistema propuesto.

La Figura 1 muestra el diseño de nuestra arquitectura para lograr dicho objetivo. A continuación, se presentan los componentes que forman el sistema:

- **Administrador:** el administrador será el encargado de dar de alta a los usuarios en la base de datos y en la *blockchain*.
- **Base de datos:** almacena los datos de los usuarios que han sido dados de alta por el administrador. Los datos registrados son el identificador (*email*), proveedor, fechas de creación y acceso, junto con el UUID (identificador único universal) del usuario.
- **Página web:** interfaz web que permite realizar acciones del sistema de ficheros descentralizado, todo de forma transparente al usuario.
- **Usuario:** el usuario se conectará a la página web para trabajar con los ficheros. Sin embargo, por temas de seguridad, antes de poder acceder a los ficheros deberá identificarse en la página web con sus credenciales, las cuales serán comprobadas en la base de datos.
- **Sistema de ficheros descentralizado:** red de nodos privados, los cuales almacenan ficheros y carpetas. Los nodos propios de cada usuario se autentican en la red mediante algoritmos de clave asimétrica (clave pública y clave privada). Una vez autenticado con éxito, pasa a participar en la red, comunicándose con el resto de nodos [8].
- ***Blockchain:*** se basa en un diseño de red permisionada, para mantener la seguridad del sistema de ficheros descentralizado. Aparte de no permitir que cualquier usuario no identificado entre y participe en la red, está dividida en canales que aíslan tanto el contrato inteligente como sus transacciones. Asimismo, almacena los metadatos, que serán todos del mismo tipo. Los campos son ID (se corresponde con el CID de IPFS), nombre del fichero y nombre del correo del usuario (unívoco para cada persona).

Para ofrecer más detalle, la Figura 2 muestra el diseño *blockchain* adoptado para esta propuesta. Como se puede observar, hay tres organizaciones, representada cada una con un color diferente. Cabe mencionar que se pueden añadir

tantas organizaciones como sean necesarias; para este caso de uso se comenzó con dos, añadiendo posteriormente una tercera. Cada organización se corresponde con una universidad u otros organismos como, por ejemplo, centros de investigación. Adicionalmente, se puede ver que se ha representado los *orderers*. Todos ellos pertenecen al mismo canal, representado con un óvalo bajo el nombre de "Channel". El canal está conectado con varios componentes, los cuales se explican seguidamente:

- **P1:** representado para cada organización. Se refiere al *peer* de la organización, que contiene una copia del libro mayor (L1) y una copia del *smart contract* (SC1). Hay que tener en cuenta que se ha decidido abstraer el diseño; para cada organización puede haber más de un *peer*, en este caso hay dos por cada una.
- **O:** dado que se refiere al nodo *orderer*, solo contiene una copia del libro mayor, no tiene copia del *smart contract*.
- **AP:** representa cada aplicación que tiene una identidad que la asocia con una organización. Por ello, se han representado las tres aplicaciones, pese a que todas contengan la misma implementación. La idea es que cada organización contenga una copia que trabaje con sus propias identidades, las cuales se corresponden con el personal trabajando en los organismos.
- **CC1:** CC o *channel configuration* contiene un registro de las organizaciones que pueden unirse a los componentes e interactuar en el canal, así como las políticas que definen la estructura de cómo se toman las decisiones y se alcanzan los resultados específicos [10]. Como muestra la Figura 2, todas las organizaciones (junto con el *orderer*), han aprobado dicha configuración, por lo que se les aplicará a cada uno de ellos.
- **CA:** representa las autoridades certificadoras que han definido las organizaciones y las identidades de sus administradores.

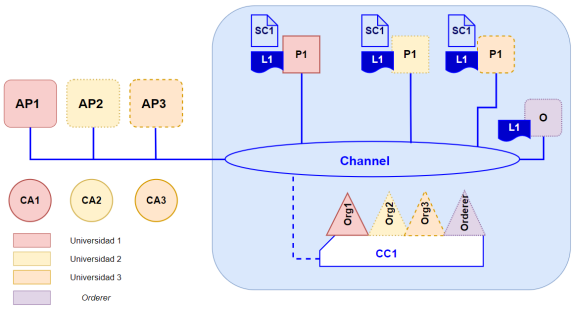

Figura 2. Diseño *blockchain* para la propuesta.

Para integrar *blockchain* e IPFS, se ha añadido la funcionalidad de enviar los metadatos creados a la *blockchain*.

El flujo consistirá en los siguientes pasos:

1. Cuando se importe uno o varios ficheros, se recogerá el nombre del fichero y el usuario que lo ha subido (será igual para la eliminación de ellos).
2. Se enviará a la *blockchain* los datos recogidos.
3. Se asociará el usuario de la página web con el de la *blockchain*.
4. Se almacenará el metadato como una transacción en la *blockchain* a través de la identidad del usuario.

Para llevar a cabo la implementación de la arquitectura, se ha utilizado la plataforma blockchain Hyperledger Fabric, React para la página web, NodeJS para la aplicación de la *blockchain*, Firebase como base de datos e IPFS para gestionar certificados.

Los beneficios de esta propuesta son la accesibilidad y trazabilidad e inmediatez de la verificación de los certificados. Dado que es un sistema descentralizado, cualquier persona con acceso al sistema puede comprobar la veracidad de la información. Esto tiene el inconveniente de la pérdida de disponibilidad en caso de caídas de red.

4. Resultados

Se realizaron varias pruebas del sistema desarrollado y se analizaron los resultados obtenidos. Entre ellas se encuentra la subida y eliminación de certificados en IPFS, con su correspondiente actualización de los metadatos en *blockchain*, tal y como se describe a continuación.

La Figura 3 muestra los nuevos certificados subidos (cert1 y cert2) en la página de IPFS, mientras que la Figura 4 se ha extraído de Postman, que es desde donde se llama a la API de la *blockchain* para obtener todos los metadatos actualmente almacenados. Como se puede observar, coinciden los datos almacenados en ambas tecnologías.

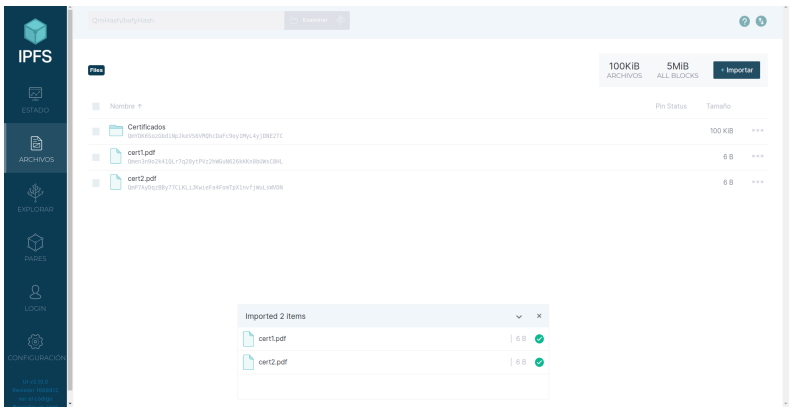

Figura 3. Archivos subidos.

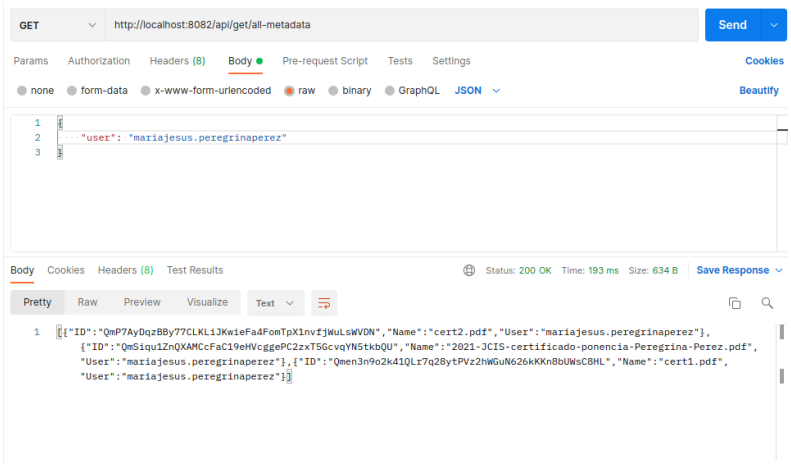

Figura 4. Metadatos en la *blockchain*.

A continuación, se muestra cómo los ficheros subidos anteriormente (cert1 y cert2) van a ser eliminados en IPFS y cómo, de forma automática, se eliminan sus metadatos de la *blockchain* para mantener la concordancia de información entre ambas tecnologías. La Figura 5 muestra los metadatos eliminados, tras haber eliminado los ficheros de IPFS. Asimismo, la Figura 6 presenta todos los registros de las pruebas que se han realizado a cabo: subida y eliminación de los certificados cert1 y cert2.

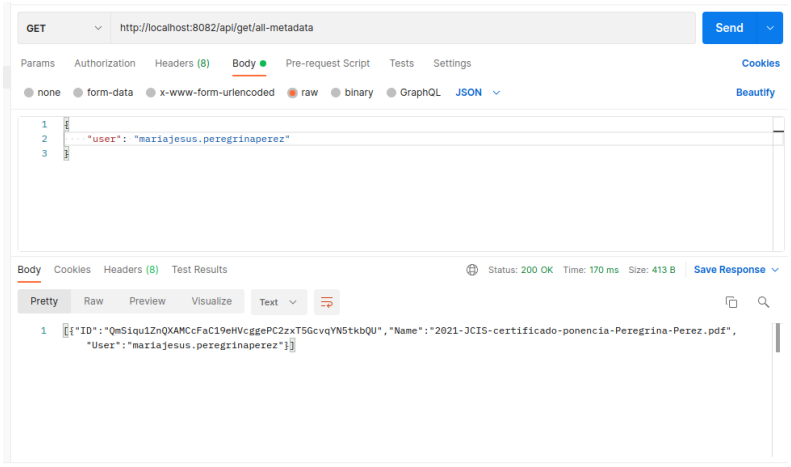

Figura 5. Metadatos tras eliminar certificados.

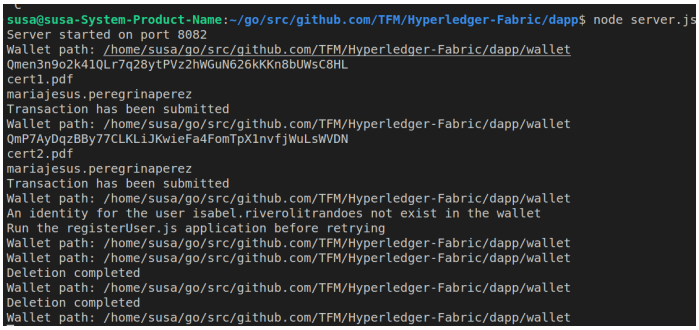

Figura 6. Registro de actividad.

5. Trabajos relacionados

Durante la revisión de la literatura no se encontraron trabajos sobre Hyperledger Fabric junto con una red privada de IPFS. En su mayoría los proyectos encontrados trabajan con Hyperledger Fabric junto con Hyperledger Composer [9]. Sin embargo, Hyperledger Composer está obsoleto desde agosto de 2021. En cambio, en nuestra propuesta se ha utilizado la API diseñada por Tam [17], junto con ejemplos de llamadas a API desde una aplicación web.

Respecto a la tecnología IPFS sí que se encontró varios trabajos relacionados [3,15]. No obstante, el más significativo es la página web de IPFS desarrollada por su propio equipo [12]. Dado que la licencia es pública, nuestra solución ha extendido este trabajo añadiendo nuevas funcionalidades, entre las que destacan el control de acceso a los ficheros, la integración con la API de *blockchain* y la gestión de la creación/eliminación de ficheros.

La propuesta más parecida a la nuestra es la presentada por Mukherjee [3]. Sin embargo, hay algunas diferencias. Entre ellas, nuestra propuesta utiliza la página web oficial de IPFS, mientras que la propuesta de Mukherjee es creada de cero. Asimismo, la propuesta de Mukherjee utiliza MongoDB, la nuestra usa Firebase. La razón para utilizar Firebase frente a MongoDB fue la sencilla API que facilita su integración, en este caso en la página web de IPFS sin perder rendimiento y seguridad para la gestión de identidades [18].

Otra propuesta parecida a nivel teórico es la presentada por Chen et al. [7]. Aunque no esté especificado si la red IPFS es pública o privada, hay un control de acceso mediante Hyperledger Fabric para trabajar con los ficheros y directorios. Mientras que esta propuesta implementa el control de acceso tanto con la página web (utilizando Firebase) como con Hyperledger Fabric.

En definitiva, que sepamos, no hay ninguna propuesta que integre Hyperledger Fabric con IPFS privado.

En la Tabla 1 se muestra la comparativa entre las diferentes propuestas mencionadas respecto a la presentada en este artículo.

Propuesta	Propósito	Blockchain	IPFS	Base de datos	Página web
Mukherjee [3]	Plataforma de vehículos	Hyperledger Fabric	No especificado	MongoDB	React
Chen et al. [7]	Gestión de archivos	Hyperledger Fabric	No especificado	-	Javascript
Hanafi et al. [9]	Gestión evidencias digitales	Hyperledger Fabric y Hyperledger Composer	Pública	-	-
Equipo IPFS [12]	Plataforma IPFS	-	Pública	-	React/Redux
Tam [17]	Implementación servidor API	Hyperledger Fabric	-	-	-
Peregrina et al.	Gestión certificados científicos	Hyperledger Fabric	Privada	Firebase	React/Redux

6. Conclusiones y trabajo futuro

Se ha desarrollado un sistema de gestión de ficheros descentralizado que, a diferencia de otros sistemas actuales, posee una adecuada trazabilidad.

Esta solución integra tecnologías novedosas como *blockchain*, IPFS, servicios REST, NodeJS, base de datos Firebase, y React para la parte web. Más específicamente, este sistema permite: (1) identificar a los usuarios antes de permitir realizar acciones sobre los ficheros; (2) importar individualmente o de forma múltiple carpetas o ficheros, tanto la creación e importación como eliminación quedarán registrados en la *blockchain*, a ese registro es lo que se conoce como metadato; y (3) dar de alta, por parte del administrador, a los usuarios tanto en la *blockchain* como en Firebase.

Gracias a este sistema, los empleados y usuarios pertenecientes a entidades institucionales, como pueden ser el PDI, PAS y estudiantado de una universidad, podrán gestionar los certificados de asistencia a eventos científicos de forma segura y descentralizada.

Como trabajo futuro se propone la mejora de los siguientes aspectos de la parte web desarrollada: hacer un seguimiento más exhaustivo de los ficheros añadiendo el método PUT (actualizar información), mejorar la eficiencia del código del contrato inteligente dado que actualmente la complejidad operacional de la búsqueda del ID de un fichero es lineal (orden N), y facilitar la gestión al administrador proporcionándole una sección única para él en la página web, en la cual pudiese dar de alta a los usuarios así como gestionarlos. Además, se pondrá a prueba la robustez de esta propuesta, evaluando su validez con personal PDI e institucional en situaciones adversas, como fallas en la red.

Referencias

1. What are smart contracts on blockchain? | IBM (2023), `https://www.ibm.com/topics/smart-contracts`
2. Androulaki, E., Barger, A., Bortnikov, V., Cachin, C., Christidis, K., De Caro, A., Enyeart, D., Ferris, C., Laventman, G., Manevich, Y., Muralidharan, S., Murthy, C., Nguyen, B., Sethi, M., Singh, G., Smith, K., Sorniotti, A., Stathakopoulou, C., Vukolić, M., Cocco, S.W., Yellick, J.: Hyperledger fabric: a distributed operating system for permissioned blockchains. In: Proceedings of the Thirteenth EuroSys Conference. pp. 1–15. ACM, Porto Portugal (Apr 2018). `https://doi.org/10.1145/3190508.3190538`
3. Arnab Mukherjee: Vehicle Platform based on Hyperledger Fabric (2023), `https://github.com/mukherjeearnab/vehicle-platform-blockchain`

4. Beduschi, A.: Digital identity: Contemporary challenges for data protection, privacy and non-discrimination rights. Big Data & Society **6**(2), 2053951719855091 (Jul 2019). https://doi.org/10.1177/2053951719855091, publisher: SAGE Publications Ltd
5. Benet, J.: What is IPFS? | IPFS Docs (Apr 2022), https://docs.ipfs.io/concepts/what-is-ipfs/#decentralization
6. Boubeta-Puig, J., Rosa-Bilbao, J., Mendling, J.: CEPchain: A graphical model-driven solution for integrating complex event processing and blockchain. Expert Systems with Applications **184**, 115578 (Dec 2021). https://doi.org/10.1016/j.eswa.2021.115578
7. Chen, J., Zhang, C., Yan, Y., Liu, Y.: FileWallet: A File Management System Based on IPFS and Hyperledger Fabric. Computer Modeling in Engineering & Sciences **130**(2), 949–966 (2022). https://doi.org/10.32604/cmes.2022.017516
8. Farmer, C.: How IPFS peer nodes identify each other on the distributed web (Jul 2018), https://medium.com/textileio/how-ipfs-peer-nodes-identify-each-other-on-the-distributed-web-8b5b6476aa5e
9. Hanafi, J., Prayudi, Y.: IPFSChain: Interplanetary File System and Hyperledger Fabric Collaboration for Chain of Custody and Digital Evidence Management. International Journal of Computer Applications **183**, 24–31 (Dec 2021). https://doi.org/10.5120/ijca2021921808
10. Hyperledger Fabric: Channels — hyperledger-fabricdocs master documentation (2023), https://hyperledger-fabric.readthedocs.io/en/release-2.3/channels.html
11. Hyperledger Fabric: The Ordering Service — hyperledger-fabricdocs master documentation (2023), https://hyperledger-fabric.readthedocs.io/en/release-2.3/orderer/ordering_service.html
12. IPFS: IPFS Web UI (Oct 2022), https://github.com/ipfs/ipfs-webui
13. Palai, A., Vora, M., Shah, A.: Empowering light nodes in blockchains with block summarization. In: 2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS). pp. 1–5 (2018)
14. Peregrina-Pérez, M.J., Lagares-Galán, J., Boubeta-Puig, J.: Hyperledger Fabric blockchain platform. In: Distributed Computing to Blockchain: Architecture, Technology, and Applications, pp. 283–295. Elsevier, London, United Kingdom, 1 edn. (2023), https://doi.org/10.1016/B978-0-323-96146-2.00014-0
15. ProtoSchool: IPFS Tutorial | Mutable File System, https://proto.school/mutable-file-system
16. Rosa-Bilbao, J., Boubeta-Puig, J., Rutle, A.: EDALoCo: Enhancing the accessibility of blockchains through a low-code approach to the development of event-driven applications for smart contract management. Computer Standards & Interfaces **84**, 103676 (Mar 2023), https://doi.org/10.1016/j.csi.2022.103676
17. Tam, K.C.: Rework: An Implementation of API Server for Hyperledger Fabric Network (Fabric v2.2) (Aug 2020), https://kctheservant.medium.com/rework-an-implementation-of-api-server-for-hyperledger-fabric-network-fabric-v2-2-a747884ce3dc
18. Tech, M.: MongoDB vs Firebase: Which Is The Best Database In 2022 (Jun 2022), https://medium.com/mqos-technologies/mongodb-vs-firebase-which-is-the-best-database-in-2022-aff873566586
19. Xu, X., Weber, I., Staples, M.: Architecture for Blockchain Applications. Springer International Publishing, Cham (2019)