ARQUITECTURA *BLOCKCHAIN* HÍBRIDA PARA EL REGISTRO DE HISTORIALES MÉDICOS

ALEJANDRO HERNÁN SON ROMERO
https://orcid.org/0009-0005-5504-3803
20202016@aloe.ulima.edu.pe
Universidad de Lima, Perú

NICOLAS XAVIER HERRERA MEDINA
https://orcid.org/0009-0009-1859-8956
20201004@aloe.ulima.edu.pe
Universidad de Lima, Perú

PABLO ALBERTO ROJAS JAEN
https://orcid.org/0000-0002-9955-6740
projas@ulima.edu.pe
Universidad de Lima, Perú

Recibido: 5 de julio del 2025 / Aceptado: 15 de octubre del 2025
doi: https://doi.org/10.26439/interfases2025.n022.8092

RESUMEN. En este trabajo se presenta una arquitectura de *blockchain* híbrida para mejorar la gestión e interoperabilidad de los historiales médicos electrónicos. Se combina una *blockchain* pública permisionada basada en Polkadot, la cual gestiona roles y permisos con una *blockchain* privada implementada en Hyperledger Fabric que, a la vez, está encargada del almacenamiento de los datos médicos de los pacientes. Los registros textuales se guardan en CouchDB, mientras que las imágenes en el IPFS se representan como *tokens* no fungibles, bajo un enfoque centrado en el paciente. Las pruebas de estrés mostraron latencias promedio de 2050 ms para la creación de historiales médicos electrónicos y 2000 ms para su intercambio, con un uso de CPU del 65 % y memoria de 170 MB, lo que evidencia eficiencia y estabilidad. La arquitectura propuesta demuestra ser una solución escalable y segura para entornos hospitalarios, ya que optimiza recursos y fortalece la confidencialidad de la información médica del paciente.

PALABRAS CLAVE: *blockchain* híbrida / historial médico electrónico / *blockchain* privada / *blockchain* pública / interoperabilidad / seguridad

HYBRID BLOCKCHAIN ARCHITECTURE FOR MEDICAL RECORD MANAGEMENT

ABSTRACT. This article introduces a hybrid blockchain architecture to enhance Electronic Medical Record (EMR) management and interoperability. It integrates a permissioned public blockchain on Polkadot—managing roles and permissions—with a private blockchain on Hyperledger Fabric responsible for EMR storage. Text data are stored in CouchDB and medical images in IPFS as Non-Fungible Tokens (NFTs), following a patient-centric model. Stress tests yielded average latencies of 2050 ms for EMR creation and 2000 ms for sharing, with 65 % CPU and 170 MB memory usage, indicating system stability and efficiency. The proposed architecture provides a scalable, secure, and interoperable solution suitable for healthcare environments that demand data confidentiality and controlled access.

KEYWORDS: hybrid blockchain / EHR / private blockchain / public blockchain / interoperability / security

INTRODUCCIÓN

A pesar de los avances tecnológicos en infraestructura para telecomunicaciones y el desarrollo de *software*, todavía persisten deficiencias para garantizar un servicio de salud de calidad a los pacientes. En este sector, el manejo de historiales médicos ha evolucionado de ser registros físicos que contienen información delicada —sobre la salud y diagnósticos— almacenados localmente, a ser historiales médicos electrónicos (HME) que se almacenan en la nube (International Organization for Standardization, 2019). Si bien esta transformación digital ha mejorado la gestión y el almacenamiento de información médica, los HME presentan problemas de interoperabilidad entre instituciones, así como un alto riesgo de ataques cibernéticos, fallos humanos y ausencia de estándares (Jayabalan & Jeyanthi, 2022; Mani et al., 2021; Miyachi & Mackey, 2021; Samala & Rawas, 2024). Estas vulnerabilidades exponen la información privada de los HME a terceros no autorizados, lo que, según Mulligan y Braunack-Mayer (2004), podría presentar un riesgo de discriminación por condiciones médicas.

Adicionalmente, la falta de acceso a un historial médico unificado puede dar lugar a la creación de múltiples versiones de este, lo que afectaría significativamente la calidad del servicio (Mulligan & Braunack-Mayer, 2004). Por ejemplo, en el 2010, en Lima (Perú), se identificó que, de 450 historiales médicos, 147 estaban incompletos y 140 se habían perdido (Montañez-Valverde et al., 2015). Asimismo, entre 2016 y 2017, se reportó un incremento del 89 % en ataques cibernéticos relacionados al sector salud en Estados Unidos (Awad Abdellatif et al., 2021), en los que 3 620 000 registros de pacientes fueron comprometidos en lo que se conoció como el mayor incidente del 2016 (Abouelmehdi et al., 2018).

Dada esta problemática, diversos autores han propuesto soluciones innovadoras basadas en tecnología de *blockchain*. Haas et al. (2011) sugirieron dar más control al paciente sobre su información, mientras que Jin et al. (2019) recomendaron aprovechar las ventajas de sistemas autorizados y sin permisos en arquitecturas de *blockchain*, y Liu et al. (2023) propusieron el uso de almacenamiento dentro y fuera de la cadena para reducir la carga del sistema. Por su parte, Mani et al. (2021) plantearon una arquitectura segura para la gestión de datos sanitarios centrada en el paciente, basada en contratos inteligentes (Cintel). En esta línea, Jayabalan y Jeyanthi (2022) propusieron un *framework* que integra la *blockchain* con el sistema de archivos interplanetario (*interplanetary file system*, IPFS), a fin de crear una arquitectura descentralizada y distribuida, y preservar la privacidad mediante un enfoque centrado en el paciente. Finalmente, Guo et al. (2019) sugirieron un sistema en el que los doctores consultan, modifican y agregan información mediante enlaces de un solo uso para almacenar los registros en nodos en el borde de la red.

Para abordar los desafíos de seguridad y confidencialidad en los registros médicos, en este trabajo se propone una arquitectura de *blockchain* híbrida (BHIB), compuesta por una *blockchain* privada (BPRIV) para la gestión de los HME y una *blockchain* pública

(BPUB) permisionada para la gestión de permisos de los usuarios. Para el almacenamiento fuera de la cadena, se utilizó el IPFS y para las imágenes los *tokens* no fungibles (*non-fungible token*, NFT). Este enfoque brinda tres ventajas fundamentales: alta privacidad de los datos, puesto que el enfoque centrado en el paciente le otorga el total dominio de sus datos a estos; alta seguridad, ya que, en adición a la seguridad de la propia *blockchain*, se emplea un almacenamiento dentro y fuera de la cadena; y mayor flexibilidad de datos, puesto que se proponen NFT para el almacenamiento de imágenes, como placas de rayos X fuera de la cadena y los datos textuales de los HME dentro de la cadena.

El presente artículo se organiza de la siguiente manera: en la sección 2, se discuten las diferentes soluciones tradicionales existentes y otras propuestas basadas en la *blockchain*; en la sección 3, se profundiza en la propuesta planteada y en la metodología empleada; en la sección 4, se detalla el proceso de implementación; en la sección 5, se presentan los resultados; en la sección 6, se hace una discusión de estos; finalmente, en la sección 7, se comparten las conclusiones.

ESTADO DEL ARTE

Tipos de almacenamiento usados en la gestión de historiales médicos electrónicos

La gestión de HME en sistemas de *blockchain* emplea tres modelos de almacenamiento —dentro de la cadena, fuera de la cadena y el modelo híbrido—, con el fin de resolver problemas de seguridad y escalabilidad. El almacenamiento dentro de la cadena, que guarda los datos directamente en ella, es cada vez menos utilizado debido a su baja disponibilidad (Jayabalan & Jeyanthi, 2022). En contraste, el almacenamiento fuera de la cadena se ha convertido en una estrategia común que mejora la escalabilidad, pues registra solo referencias criptográficas en la *blockchain*. Investigadores como Dagher et al. (2018), Hussien et al. (2021), Nhan et al. (2024) y Kang et al. (2022) utilizan IPFS para este fin, almacenando los HME o sus *hashes*; sin embargo, este enfoque introduce nuevas preocupaciones de seguridad y dependencia de plataformas externas.

Para solucionar estas preocupaciones, el almacenamiento híbrido —que combina el almacenamiento dentro y fuera de la cadena— equilibra seguridad, escalabilidad y eficiencia (Amanat et al., 2022; Chelladurai et al., 2021; Kaur et al., 2023; Lee et al., 2022; Samala & Rawas, 2024; Tanwar & Thakur, 2023). En este contexto, estudios como los de Jayabalan & Jeyanthi (2022) y Mani et al. (2021), que se resumen en la Tabla 2.1, muestran cómo el uso del IPFS contribuye a abordar los problemas de escalabilidad y almacenamiento de grandes volúmenes de datos médicos. En ambos casos, los investigadores almacenan los HME fuera de la cadena y preservan dentro de la *blockchain* los *hashes* o metadatos cifrados, lo que fortalece la integridad de los registros y permite implementar múltiples capas de seguridad mediante cifrado.

Del mismo modo, Rajput et al. (2021), Haddad et al. (2024) y Uddin et al. (2021) emplearon Cintel para gestionar los accesos y las transacciones, a fin de guardar los registros de auditoría dentro de la cadena, mientras que los HME permanecen fuera de ella. Por su parte, Shen et al. (2019) (Tabla 2.1) han propuesto una red *peer-to-peer* externa en lugar de un IPFS y así conservar dentro de la cadena únicamente los metadatos y *hashes* de los historiales. Estos ejemplos ilustran la diversidad de estrategias *off-chain* que buscan mantener la trazabilidad sin comprometer la eficiencia del sistema.

Como se detalla en la Tabla 1, el enfoque híbrido logra combinar seguridad y escalabilidad, pero introduce nuevos retos relacionados con la gestión de claves criptográficas y la integridad de los datos fuera de la cadena. Sobre la base de lo visto en el estado del arte sobre el almacenamiento, este trabajo propone una arquitectura de *blockchain* híbrida para la gestión de HME debido a su capacidad de equilibrar las demandas de seguridad, escalabilidad y eficiencia. Este enfoque híbrido permite almacenar los metadatos y las referencias a los datos médicos en la *blockchain*, lo que asegura su trazabilidad e integridad mediante la BPRIV, mientras que los HME de mayor tamaño —como las imágenes y documentos médicos— se almacenan fuera de la cadena en IPFS. Esto garantiza que la *blockchain* no se sobrecargue, mejora la escalabilidad del sistema y asegura que los pacientes y médicos puedan acceder a los datos de manera eficiente y segura a través de los Cintel que gestionan los accesos.

Tipos de *blockchain* utilizados en la gestión de historiales médicos electrónicos

Los tres tipos principales de *blockchain* que se utilizan en la gestión de HME son la BPUB, la BPRIV y la BHIB. Cada una de estas opciones ha sido explorada en distintos estudios para abordar desafíos relacionados con interoperabilidad, seguridad y escalabilidad.

El uso de la BPUB ha sido propuesto en diversos trabajos, tales como los de Fatokun et al. (2021), Hossain Faruk et al. (2021), Kumari et al. (2024), Mauricio et al. (2024), Puneeth y Parthasarathy (2024) y Wang et al. (2021). Asimismo, Mandarino et al. (2024) y Omar et al. (2021) —mencionados en la Tabla 1— emplearon Ethereum como BPUB para garantizar la descentralización y la transparencia. En particular, Mandarino et al. (2024) integraron computación en el borde (*edge computing*) para almacenar datos de manera local en los dispositivos, lo que redujo la dependencia de servidores centralizados. Por su parte, Omar et al. (2021) priorizaron la seguridad mediante el cifrado de HME en la nube, empleando Cintel para regular el acceso. De modo similar, Zhang et al. (2022) plantearon un modelo en el que la BPUB facilita la trazabilidad de los datos médicos, utilizando protocolos de pago justo que incentivan la participación de los pacientes. Aunque la BPUB ofrece ventajas en trazabilidad y transparencia, enfrenta limitaciones notables de escalabilidad y eficiencia (Tabla 1).

En contraste, el uso de la BPRIV ha sido explorado para reforzar la seguridad, privacidad y control de acceso en la gestión de HME (Abunadi & Kumar, 2021; Ali, Rahim,

Pasha et al., 2021; Ali, Rahim, Ali, et al. 2021; Antwi et al., 2021; Guo et al., 2019; Hashim et al., 2022; Huang et al., 2021; Selvarajan & Mouratidis, 2023; Singh et al., 2021; Wu et al., 2022). En particular, Awad Abdellatif et al. (2021) —mencionados en la Tabla 1— presentaron MEdge-Chain, una BPRIV combinada con computación en el borde para gestionar de manera eficiente grandes volúmenes de HME distribuidos en múltiples hospitales. Este sistema permite la monitorización automatizada y la detección temprana de eventos médicos críticos, ejecutando transacciones a través de Cintel. Además, existen BPRIV de consorcio (Exceline & Nagarajan, 2024; Li et al., 2024; Liu et al., 2023; Mohey Eldin et al., 2023; Xiao et al., 2021), las cuales son *blockchains* permisionadas administradas por un grupo de organizaciones o entidades, lo que optimiza el control de acceso y la privacidad, aunque con limitaciones de interoperabilidad, como se aprecia también en la Tabla 1.

Finalmente, la BHIB combina características de *blockchains* públicas y privadas, ofreciendo una solución equilibrada frente a los desafíos de seguridad, escalabilidad e interoperabilidad en la gestión de los HME. Algunos ejemplos representativos se encuentran en Uppal et al. (2023) y Chelladurai y Pandian (2022), quienes, como se detalla en la Tabla 1, emplearon Hyperledger Fabric (HF) para gestionar permisos de acceso, compartir datos entre proveedores y permitir a los pacientes controlar sus propios historiales. Asimismo, Kaur et al. (2023) integraron BPUB y BPRIV en el contexto de la cirugía médica, empleando la primera para garantizar la inmutabilidad de los datos y la segunda para proporcionar acceso controlado a los pacientes autorizados.

Como se observa en la Tabla 1, el enfoque híbrido representa una síntesis de las ventajas de ambos tipos de *blockchain*, lo que resuelve los problemas de escalabilidad de las BPRIV y los de seguridad de las BPUB. Entonces, sobre la base de lo discutido en el estado del arte, y tal como ya se ha mencionado, este trabajo propone una arquitectura BHIB para la gestión de los HME. Esta solución permite superar las limitaciones de escalabilidad y eficiencia de las BPUB, a la vez que aborda los desafíos de interoperabilidad y seguridad propios de las BPRIV. En la arquitectura propuesta, la BPUB se encarga de la verificación de accesos y la interoperabilidad interhospitalaria, mientras que la BPRIV gestiona los permisos de acceso a los HME, lo que garantiza que solo los actores autorizados puedan modificar los datos. Este enfoque asegura un manejo eficiente de los HME, por lo que logra ofrecer privacidad, seguridad y control de acceso, sin comprometer la interoperabilidad entre instituciones.

Tabla 1

Comparativa de los trabajos más resaltantes para sistemas de historiales médicos electrónicos

Autor	Metodología	Tecnología de almacenamiento	Seguridad y privacidad	Resultados	Limitaciones
Shen et al. (2019)	Diseño de un sistema de BPUB para el intercambio de datos médicos	Dentro de la cadena y fuera de la cadena (BPUB)	Control de acceso mediante *blockchain*, pero con limitaciones en escalabilidad y latencia	Implementación de BPUB para compartir datos médicos y lograr mayor trazabilidad y seguridad	Problemas de escalabilidad y latencia en la red pública
Mandarino et al. (2024)	Integración de *blockchain* con computación en el borde (*edge computing*) para manejar grandes volúmenes de datos médicos	Fuera de la cadena (*edge computing* para dispositivos IoT)	Mejora de la privacidad al procesar datos cerca de su origen y garantiza seguridad mediante Cintel	Eficiencia en el manejo de grandes volúmenes de datos en tiempo real y reducción de la latencia	Abordaje inadecuado de la interoperabilidad con otros sistemas de salud
Awad Abdellatif et al. (2021)	Implementación de un sistema híbrido que integra *blockchain* con *edge computing* para la gestión eficiente de datos médicos	Fuera de la cadena (*edge computing* e IPFS)	Garantía de privacidad al descentralizar el procesamiento de datos y manejar accesos mediante Cintel	Mejora de la escalabilidad y eficiencia al integrar *blockchain* con *edge computing* y IPFS	Complejidad en la sincronización de datos entre nodos y dispositivos IoT
Uddin et al. (2021)	Uso de HF para gestionar de manera segura y eficiente los accesos a los HME	Dentro de la cadena (HF) y fuera de la cadena (base de datos externa)	Gestión granular de accesos y transacciones mediante Cintel	Seguridad robusta en la gestión de HME mediante HF (eficiente para instituciones de salud privadas)	Limitada interoperabilidad con otras instituciones de salud que no utilicen HF
Omar et al. (2021)	Propuesta de un sistema híbrido que combina *blockchain* y almacenamiento en la nube para gestionar los HME	Fuera de la cadena (IPFS y almacenamiento en la nube)	Encriptado de datos fuera de la cadena y manejo de claves dentro de la cadena para la protección contra accesos no autorizados	Alta transparencia en el acceso a los datos médicos y escalabilidad mediante almacenamiento fuera de la cadena	Riesgos de seguridad inherentes a la dependencia en la nube para almacenamiento fuera de la cadena

(*continúa*)

(*continuación*)

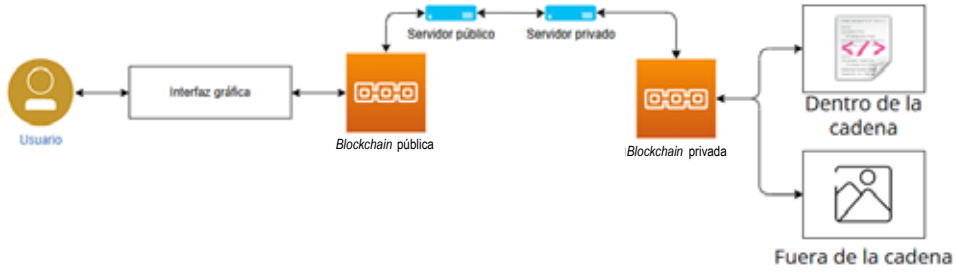
Autor	Metodología	Tecnología de almacenamiento	Seguridad y privacidad	Resultados	Limitaciones
Mani et al. (2021)	Desarrollo de un sistema basado en BPRIV para gestionar HME mediante almacenamiento en IPFS	Fuera de la cadena (IPFS)	Gestión de permisos de acceso mediante Cintel en HF	Optimización del uso de IPFS para el almacenamiento de grandes volúmenes de datos médicos, con control de acceso a través de BPRIV	Interoperabilidad limitada con otros sistemas de salud al depender de una BPRIV
Jayabalan y Jeyanthi (2022)	Modelo de BHIB para gestionar HME mediante almacenamiento distribuido en IPFS	Fuera de la cadena (IPFS)	Gestión de claves y permisos mediante Cintel para asegurar el acceso a los datos médicos	Escalabilidad mejorada mediante el uso de IPFS para almacenamiento distribuido de datos grandes, como imágenes médicas	Seguridad de los datos fuera de la cadena que depende de la solidez de IPFS y el manejo de claves criptográficas
Rajput et al. (2021)	Propuesta de un marco de BPRIV para el intercambio seguro de datos médicos encriptados	Dentro de la cadena (BPRIV)	Enfoque en la privacidad mediante el uso de encriptación avanzada y control de acceso descentralizado	Seguridad mejorada mediante encriptación avanzada y control granular sobre el acceso a los HME	Baja escalabilidad y limitada interoperabilidad debido al uso exclusivo de BPRIV
Chelladurai y Pandian (2022)	Sistema automatizado de gestión de HME basado en Cintel y BPRIV	Dentro de la cadena (BPRIV)	Control de acceso mediante contratos inteligentes para garantizar la privacidad de los datos médicos	Automatización del control de acceso a HME, lo que mejora la seguridad y la eficiencia del sistema	Limitada interoperabilidad al depender de una BPRIV
Zhang et al. (2022)	Sistema híbrido de *blockchain* para preservar la privacidad de los HME mediante el almacenamiento en IPFS	Fuera de la cadena (IPFS)	Encriptado de datos médicos antes de ser almacenados fuera de la cadena, mientras los permisos y claves se gestionan dentro de la cadena	Mejora de la privacidad mediante el almacenamiento en IPFS y la gestión de claves en la *blockchain*, con alta escalabilidad	Riesgo en la dependencia de IPFS para el almacenamiento fuera de la cadena y posible complejidad en la gestión de claves criptográficas

METODOLOGÍA

Como se observa en la Figura 1, por medio de la interfaz gráfica, los usuarios interactúan con la arquitectura. En la BPUB se encuentra el Cintel que asigna roles a los usuarios, los permisos de visualización y los datos básicos de las cuentas de los usuarios. En la BPRIV se encuentra la información de los HME de los pacientes, la cual verifica la acción que el usuario solicita mediante el rol y permiso correspondiente. Una vez realizada la validación, la información solicitada retorna a la BPUB o guarda los cambios realizados. De esta manera, la lógica de las cuentas y los HME asociados a cada cuenta están separados, lo que permite a las *blockchains* estar centralizadas en una función en particular.

Figura 1

Arquitectura de la propuesta

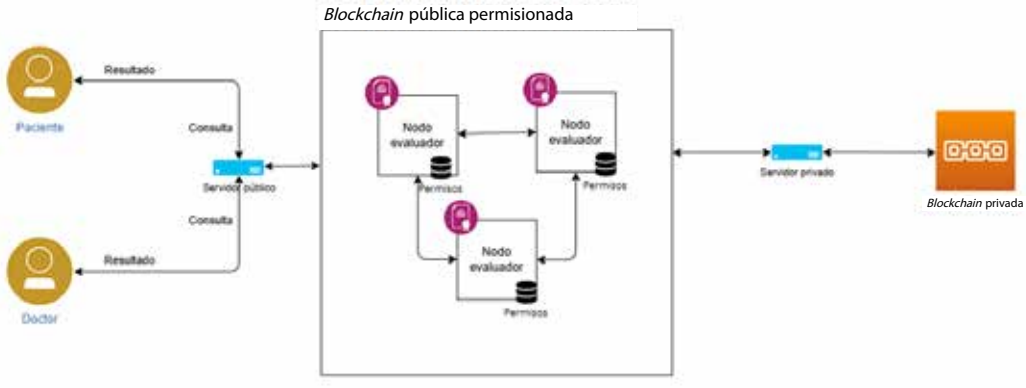


Blockchain pública permisionada

En las *blockchain* públicas permisionadas (BPUBPER) se permite que cualquier usuario que se una a la red tenga conocimiento de la identidad de los demás usuarios; sin embargo, su capacidad de acción está restringida a los privilegios otorgados según el rol designado. En la Figura 2, se observa que los usuarios pueden interactuar con la arquitectura mediante el servidor de la parte pública y este servidor. A través del Cintel "Control de acceso", se le asigna a cada usuario un rol, el cual puede ser de doctor o de paciente; de esta manera, al ser paciente, se puede garantizar o revocar permisos a doctores. Asimismo, el servidor de la parte pública se puede comunicar con el servidor de la parte privada para consultar la información de los HME.

Figura 2

Arquitectura de la blockchain *pública permisionada*

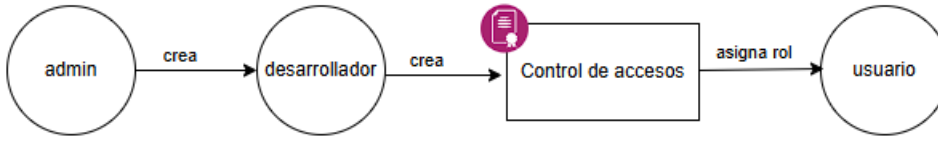


Cuentas de desarrollador y administrador

En la Figura 3, se observa cómo se genera una cuenta de desarrollador con la capacidad de crear Cintel, utilizando la cuenta de administrador de la *blockchain* que cuenta con los permisos más avanzados. A partir de esta cuenta, se establece el Cintel "Control de acceso". Este contrato se encarga de asignar roles a los usuarios, en caso de que su clave secreta esté asociada a un centro médico. Esto determina si el usuario es un doctor, en el caso de que su clave privada está registrada en la red, o si es un paciente, en el caso de que no contar con información disponible. Una vez que se tenga el rol asociado a la cuenta, el usuario podrá hacer uso de las funciones para visualizar el HME y otorgar o revocar permisos de visualización. Además, se usará el Cintel "Control de acceso" para que la BPRIV verifique los roles de quien hace la consulta, así como los permisos de visualización.

Figura 3

Diagrama de cuentas de desarrollador



Control de acceso y permisos

De acuerdo con lo mencionado, el control de acceso tiene como principal funcionalidad asignar un rol específico a cada usuario que se conecte a la red. Con este, se le asignan

los permisos respectivos. En la Tabla 2, se detallan las especificaciones de cada rol. En el caso de que se decida crear una cuenta, el usuario tendrá que usar la clave privada para crearla en la BPUBPER. Luego, ingresa su información personal y la asocia al rol de doctor o paciente, respectivamente. En caso de que el usuario posea una cuenta creada como doctor y desee crear una cuenta como paciente o viceversa, tendrá que asociar su cuenta con la información.

Tabla 2

Asignación de rol

Rol	Motivo	Consecuencia	Funcionalidades
Doctor	La clave privada se encuentra presente en una organización de la BPRIV	Se crea su cuenta y se asocia con su información personal	Modificar el HME, solicitar acceso de visualizar el HME y visualizar el HME de un paciente
Paciente	No existe una cuenta como paciente previamente	Se crea su cuenta y se asocia con su información personal	Visualizar el HME, otorgar y revocar permiso de visualizar el HME

En la Tabla 3, se observan los casos de asignación y revocación de permisos. En el primer caso, un doctor solicita acceso de visualización a un paciente, quien puede aceptar o rechazar dicha solicitud. En el segundo caso, el paciente quiere revocar el permiso a un doctor con acceso de visualización, se realiza la solicitud mediante el servidor de la parte pública y se revoca el permiso. El último caso, implica que un paciente se atendió en un centro médico que no pertenece a la arquitectura híbrida propuesta, por lo que el doctor no tiene acceso al HME de su paciente; entonces, el paciente le muestra su HME. Se utiliza el enfoque centrado al paciente, debido a que el paciente tiene el control de acceso de su HME. Esto incrementa la confiabilidad y seguridad de la información sensible.

Tabla 3

Asignación de permisos

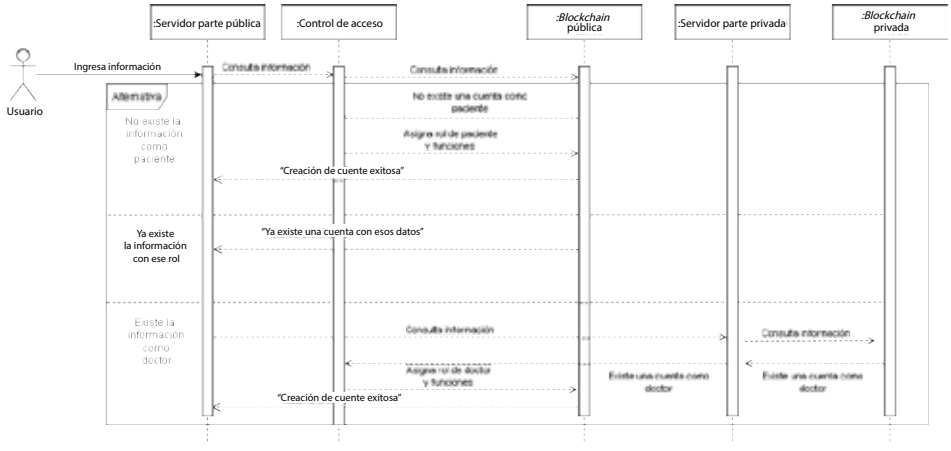
Caso	Acción del paciente
Un doctor perteneciente a una organización quiere visualizar el HME de un paciente	El paciente puede otorgar el permiso o rechazar la solicitud
El paciente quiere revocar el permiso de visualización del HME	El paciente dueño del HME puede revocar el permiso
Un doctor no perteneciente a una organización quiere visualizar el HME de un paciente	El paciente dueño del HME puede mostrarle desde su cuenta su HME

Funcionamiento de control de acceso

En la Figura 4, el usuario ingresa su información por medio del servidor público para crear una cuenta. Al registrarse, se crea de forma automática una cuenta en la red BPUBPER que le sirve para conectarse posteriormente. Cuando se suben los datos, el Cintel verifica en la *blockchain* si estos están asociados a un rol (doctor o paciente). En caso de que no lo estén, significa que es un usuario que recién está ingresando al sistema, por lo que se le otorga el rol de paciente y se le otorgan las funciones correspondientes. Si se trata de un doctor, el servidor de la parte pública se comunica con el servidor de la parte privada para verificar si su documento de identidad es de un doctor. A continuación, si se verifica su estatus, se le asigna el rol de doctor y las funcionalidades correspondientes. En el caso de que la información se encuentre en la *blockchain* y se intente crear nuevamente una cuenta con la misma información como doctor, se rechaza; como paciente, se permite guardar. Una persona solo puede tener dos cuentas: una de doctor y otra de paciente. Esta restricción existe porque un usuario podría ser doctor en un hospital y luego paciente en otro, o ser primero paciente y luego doctor.

Figura 4

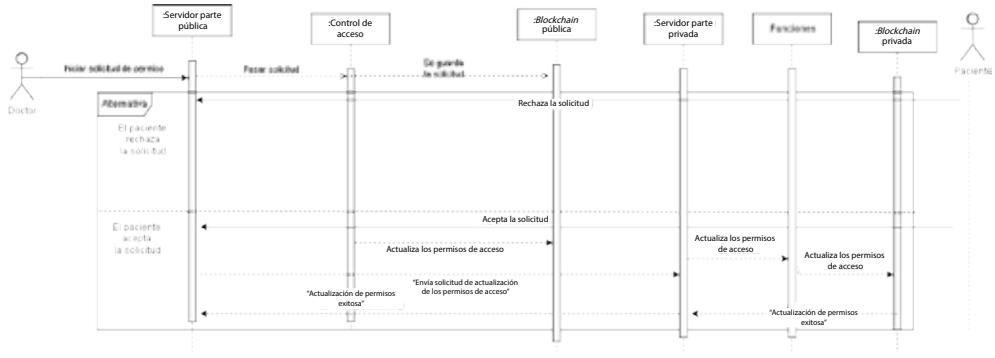
Diagrama de secuencia del control de acceso



Para realizar el control de acceso, se emplea el Cintel "Control de acceso". Este asigna el rol al usuario y las funcionalidades que aparecen en la Tabla 2. De igual manera, como se muestra en la Figura 5, la solicitud del doctor pasa por el servidor de la parte pública. Por medio del Cintel "Control de acceso", se registra la solicitud en la BPUBPER. Luego, el paciente puede decidir si rechaza la solicitud o la acepta. Si la rechaza, los permisos no se cambian. En caso contrario, se actualizan los permisos de visualización de la BPUBPER, mientras que los permisos en la BPRIV se realizan por medio del Cintel "Funciones". Finalmente, se le informa al doctor que se aceptó su solicitud.

Figura 5

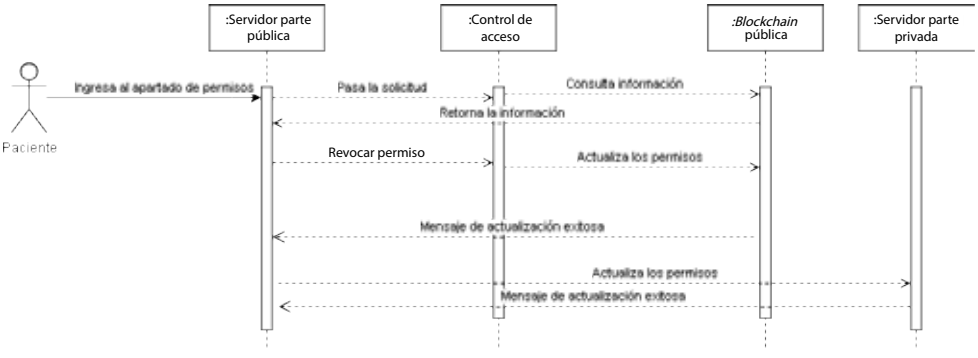
Otorgar permisos de acceso al HME



La Figura 6 detalla que el paciente puede revocar los permisos asignados a los doctores, gracias al Cintel "Control de acceso". El paciente, por medio del servidor de la parte pública, revoca los permisos del doctor y los actualiza en la BPRIV.

Figura 6

Revocar permiso de visualización a terceros



Blockchain privada

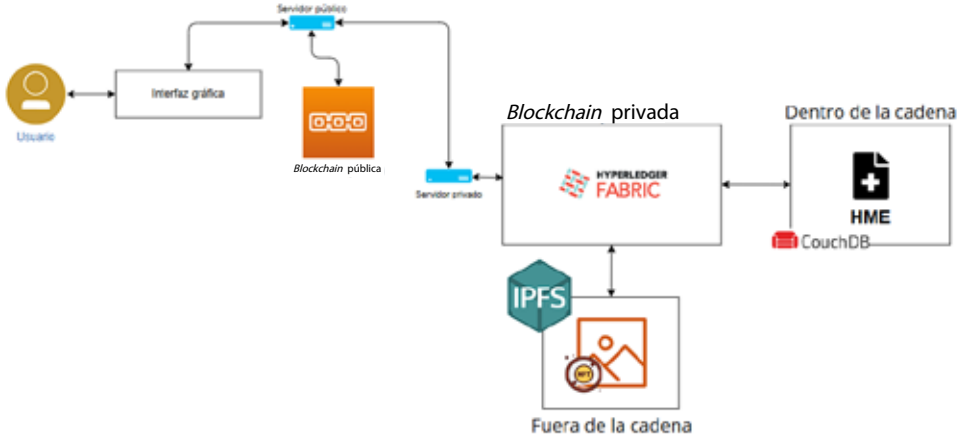
La creciente necesidad de gestionar HME de manera segura, eficiente y centrada en el paciente ha impulsado la adopción de tecnologías avanzadas como la *blockchain*. En este contexto, HF se presenta como una solución robusta para implementar una BPRIV que permita almacenar, acceder y modificar HME con altos estándares de seguridad y privacidad.

Arquitectura de la blockchain privada

Como se visualiza en la Figura 7, la BPRIV está diseñada para soportar la gestión descentralizada de HME entre dos organizaciones, pues representa a diferentes hospitales dentro de una cadena hospitalaria. Cada organización actúa como un nodo evaluador, responsable de validar transacciones y mantener una copia actualizada del libro mayor distribuido. Esta configuración permite una colaboración fluida y segura entre los diferentes hospitales, lo que garantiza la disponibilidad y consistencia de los HME. En este caso, dichos hospitales se encuentran dentro de la red de HF, mientras que los HME se encuentran divididos en el almacenamiento dentro y fuera de la cadena. Dentro de la cadena, se encuentra la información textual de los HME y el *hash* respectivo de los NFT relacionados a cada HME. Por otro lado, fuera de la cadena, se encuentran las imágenes convertidas en NFT para garantizar su trazabilidad.

Figura 7

Visualización del almacenamiento de un HME



Organizaciones y nodos

Cada hospital perteneciente a la BPRIV se configura como una organización independiente. Los nodos evaluadores de cada organización son responsables de validar transacciones, ejecutar el Cintel y mantener la integridad del libro mayor. Esta estructura permite una gestión distribuida y segura de los HME, en el que cada nodo tiene la capacidad de verificar y registrar transacciones, lo que asegura la transparencia y confiabilidad del sistema.

Contratos inteligentes

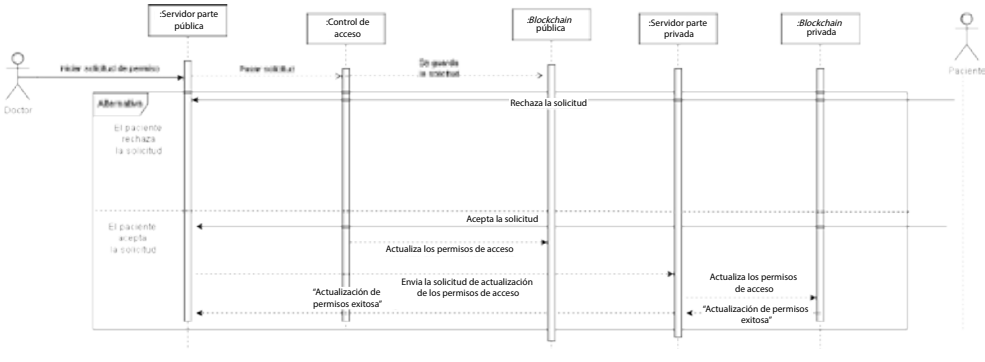
Los Cintel —conocidos en HF como códigos de cadena— son fundamentales para gestionar las operaciones sobre los HME. Estos Cintel definen las reglas y procedimientos para la creación, almacenamiento, actualización y acceso a los registros médicos. A continuación, se describen algunas de sus funciones clave.

- Creación del HME: permite registrar un nuevo HME en la BPRIV. Esta función toma los datos del paciente y crea un registro único en CouchDB.
- Modificación del HME: facilita la modificación de HME existentes. Solo los usuarios con permisos adecuados pueden actualizar la información, lo que garantiza la integridad y exactitud de los datos.
- Visualización del HME: permite que los usuarios visualicen los HME que tienen acceso.
- Verificación de permisos: antes de permitir cualquier operación sobre un HME, esta función verifica los permisos del usuario solicitante y asegura que solo los usuarios autorizados puedan acceder o modificar los datos.
- Agregación y eliminación de hashes de imágenes: permite añadir y eliminar los hashes de las imágenes generadas por IPFS, lo que modifica el HME.

Como se muestra en la Figura 8, el doctor, tras enviar su solicitud al servidor público, puede visualizar el HME de un paciente. Esto se comprueba utilizando el Cintel "Control de acceso", si el usuario tiene permiso de visualización. En el caso de que lo tenga, se retorna el historial médico; de lo contrario, no podrá visualizarlo y deberá solicitar el acceso al paciente.

Figura 8

Visualización de historial médico electrónico de un tercero



Almacenamiento de datos

Como se puede apreciar en la Figura 7, se plantea una solución de almacenamiento híbrida que combina las ventajas del almacenamiento dentro y fuera de la cadena:

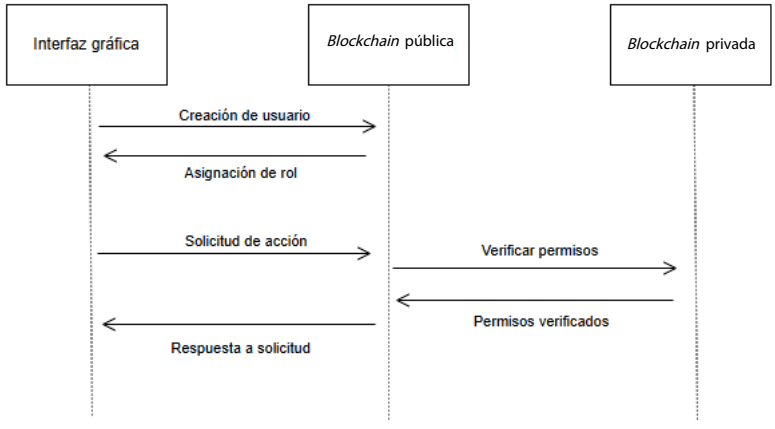
- Dentro de la cadena: utilizada para almacenar la parte textual de los HME directamente en la blockchain. Asimismo, se almacenan los hashes que vinculan a los archivos fuera de la cadena, lo que garantiza la integridad de los datos y su recuperación eficiente.
- Fuera de la cadena: la información visual crítica, como radiografías y tomografías, se almacena fuera de la cadena como NFT.

Verificación de permisos

Como se visualiza en la Figura 9, cuando un usuario intenta acceder o modificar un HME, la BPRIV verifica sus permisos consultando los roles y autorizaciones almacenados en la BPUBPER. Este proceso de verificación incluye la autenticación del usuario y la confirmación de sus permisos específicos. Solo los usuarios con los permisos adecuados pueden proceder con la visualización o modificación de los HME, para asegurar que los datos se mantengan protegidos y privados.

Figura 9

Verificación de permisos



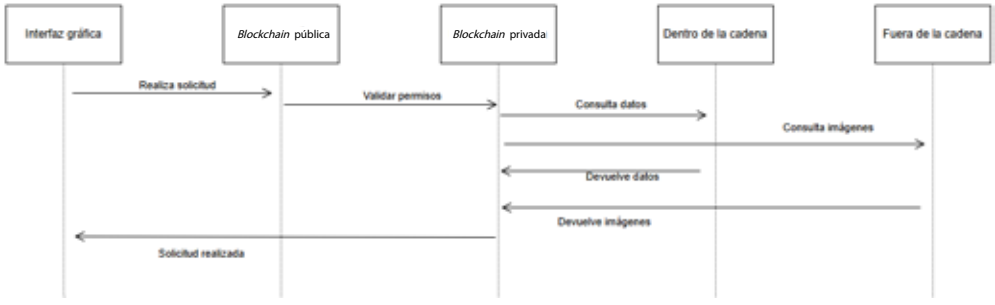
Flujo de trabajo de consulta de HME

Como se aprecia en la Figura 3.7, para consultar un HME, el usuario envía una solicitud a través de la interfaz gráfica. La BPUBPER la recibe y valida los permisos respectivos con la BPRIV, y esta verifica los permisos del usuario consultando a la BPUBPER. Si los permisos son válidos, la BPRIV recupera el HME desde el almacenamiento dentro y

fuera de la cadena y retorna la información solicitada a la interfaz gráfica. Este proceso asegura que solo los usuarios autorizados puedan acceder a los datos médicos, manteniendo la privacidad y seguridad de la información.

Figura 10

Flujo de trabajo de consulta de HME

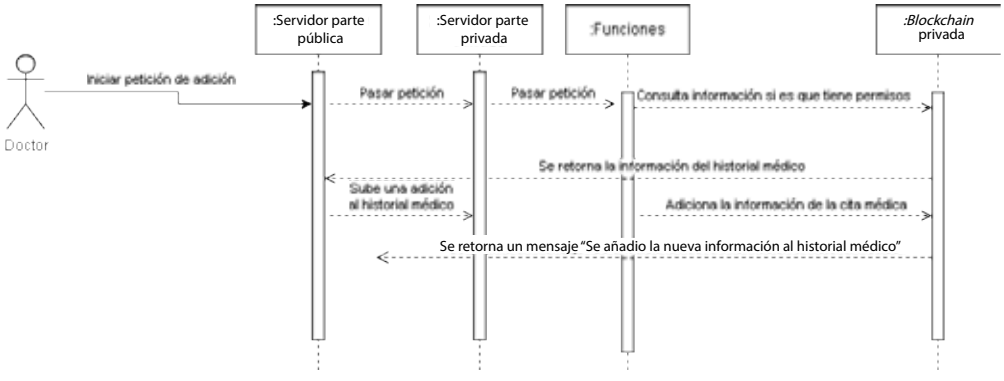


Flujo de trabajo de modificación de HME

Como se visualiza en la Figura 11, para modificar un HME, el usuario envía una solicitud de modificación a través del servidor de la parte pública. La BPRIV verifica los permisos del usuario consultando a la BPUBPER. Si los permisos son válidos, los cambios se aplican y el HME se actualiza en la base de datos dentro de la cadena. Los cambios también pueden involucrar actualizaciones de *hashes* que apuntan a nuevas versiones de archivos en la base de datos fuera de la cadena. Este flujo de trabajo garantiza que solo los usuarios autorizados puedan modificar los datos médicos, lo que preserva la integridad y exactitud del HME.

Figura 11

Flujo de trabajo de modificación de HME



Mecanismos y seguridad

La BPRIV implementa varios mecanismos de seguridad para proteger los datos médicos, incluyendo:

- Cifrado de datos: los datos se cifran tanto en tránsito como en reposo para protegerlos contra accesos no autorizados.
- Autenticación de usuarios: se utiliza autenticación basada en certificados digitales para asegurar que solo los usuarios autorizados puedan acceder al sistema.
- Políticas de control de acceso: las políticas de control de acceso detalladas aseguran que solo los usuarios con los permisos adecuados puedan realizar operaciones específicas sobre los HME.

Privacidad de los datos

La privacidad de los HME se garantiza mediante un enfoque centrado en el paciente. Los datos médicos solo son accesibles para los usuarios que fueron autorizados explícitamente por el paciente. Los *hashes* en CouchDB, que apuntan a datos en IPFS, aseguran que los datos almacenados fuera de la cadena sean recuperables de manera segura y eficiente. Este enfoque asegura que la privacidad del paciente se mantenga en todo momento.

Interconexión con la *blockchain* pública permisionada

La arquitectura se fundamenta en una estricta separación de responsabilidades: la BPUBPER gestiona exclusivamente los roles y permisos de los usuarios, mientras que la BPRIV se dedica al almacenamiento y manejo de los datos médicos (HME). Esta especialización optimiza la seguridad y eficiencia del sistema.

La interconexión se materializa a través de dos servidores dedicados. El servidor público gestiona las peticiones del usuario, comunicándose con la BPUBPER a través del Cintel "Control de acceso". Para validar cualquier solicitud, el servidor privado, que interactúa con la BPRIV mediante el Cintel "Funciones", consulta al servidor público para verificar los permisos correspondientes en la BPUBPER. Este flujo garantiza que solo usuarios autorizados interactúen con los HME.

IMPLEMENTACIÓN

Se realizaron las pruebas en computadoras con el sistema operativo Windows 11 Enterprise. Con respecto al *hardware*, se utilizó un procesador 12th Gen Intel (R) Core (TM) i7-12700 2,10 GHz, con 64 GB (63,7 GB usable) con una GPU Nvidia GeForce RTX 3080 de 10 GB.

Configuración del entorno de desarrollo

El entorno de desarrollo se configuró integrando una BPRIV sobre HF y una BPUBPER sobre Polkadot. La implementación de HF requirió Docker, Docker Compose y sus herramientas CLI para configurar una red de dos nodos y un canal. Para el almacenamiento de datos, esta red se conectó con CouchDB (textuales) e IPFS (no textuales, vinculados por *hash*). Paralelamente, se desplegó el nodo de Polkadot. La comunicación entre ambas *blockchains* se gestionó mediante servidores desarrollados en Node.js, mientras que Postman se utilizó como interfaz para ejecutar y validar las solicitudes.

Desarrollo de contratos inteligentes

El desarrollo de Cintel en HF implica la creación de código de cadena en Go para la gestión de HME, control de acceso y permisos. Estos contratos se despliegan en la red de HF y se someten a pruebas unitarias para asegurar su funcionalidad.

En el caso de Polkadot, se utiliza el *framework* Substrate para desarrollar Cintel que gestionan la asignación y verificación de roles y permisos de usuarios. Los contratos se despliegan en la red Polkadot y se realizan pruebas de integración con HF para garantizar una comunicación fluida.

El Cintel "Control de acceso" presenta once funciones: add_user, assign_role, user_exists, user_role, get_role, grant_permission, revoke_permission, has_permission, get_grantees, approve_access y request_access. De todas ellas, las únicas con las que interactúa el usuario directamente son add_user, assign_role, grant_permission y revoke_permission, ya que permiten registrar la cuenta, elegir el rol y otorgar o revocar permisos.

Integración de componentes

La arquitectura integra HF con CouchDB para gestionar los datos textuales de los HME. Paralelamente, IPFS almacena los datos no textuales, mientras sus *hashes* se registran en CouchDB para que las transacciones de HF puedan recuperar y modificar el registro completo.

Blockchain pública permisionada

Como se muestra en la Figura 2, la BPUBPER utiliza Polkadot en un despliegue local, basado en la plantilla de una red local de Substrate (Ink, s. f.), modificada para ser permisionada y compatible con Cintel desarrollados en ink! (Ink, s. f.). Se eligió Polkadot por cumplir con los requerimientos mínimos: rápido crecimiento, amplia documentación y facilidad para crear Cintel. El servidor público está implementado con Polkadot JS API, Express JS y Node JS.

Interconexión

La comunicación se hace entre el servidor de la parte pública con el de la privada. Cada uno de estos servidores se comunica con sus *blockchains* correspondientes.

DISEÑO EXPERIMENTAL

El conjunto de datos (*dataset*) utilizado para las pruebas fue "EHRXQA: A Multi-Modal Question Answering Dataset for Electronic Health Records with Chest X-Ray Images" (Bae et al., 2023). Este está compuesto por la combinación de dos *dataset*: MIMIC-CXR-VQA y EHRSQL (MIMIC-IV). El primero de ellos es un recurso integral diseñado para tareas de respuesta a preguntas visuales (*visual question answering*, VQA) en el ámbito médico, centrado principalmente en radiografías de tórax. Además, es derivado principalmente de los *datasets* MIMIC-CXR-JPG y Chest ImaGenome, obtenidos de Physionet. El segundo es una base de datos que contiene registros de salud electrónicos de pacientes, así como la información demográfica, los resultados de pruebas médicas, los diagnósticos, los procedimientos médicos, las medicaciones administradas, entre otros.

Asimismo, el *dataset* está estructurado en archivos JSON. Esto facilita su manipulación, análisis y transferencia a la BPRIV de HF, donde su integridad y seguridad se garantizan mediante Cintel y políticas de control de acceso.

Se realizaron pruebas de las funciones de los Cintel utilizando un fragmento del *dataset* previamente mencionado. Estas pruebas evaluaron métricas de uso de recursos, latencia, velocidad de consulta *off-chain* y costo por función (Mani et al., 2021; Singh et al., 2021). Para las mediciones se empleó un código presente en el servidor, que facilita la comunicación con la red pública.

Es importante señalar que, en BPUB, la autosostenibilidad se logra mediante recompensas a los nodos que realizan tareas específicas. Por ejemplo, en Bitcoin, los mineros reciben recompensas por validar bloques, mientras que en Ethereum y EOSIO, los validadores obtienen compensaciones por la creación de bloques de transacciones. Cada función ejecutada mediante Cintel genera una comisión que varía según la *blockchain* utilizada, además de poder dar propinas para incrementar la prioridad de la transacción enviada (Polkadot, 2024a).

Pruebas de estrés

Para evaluar el rendimiento de la BPUBPER y realizar pruebas de estrés, se enviaron mil peticiones en tres modalidades: concurrente, secuencial y por lotes (Tabla 4). Cabe destacar que tanto la BPRIV como la BPUBPER, junto con los servidores respectivos, se ejecutan en la misma máquina.

Tabla 4

Pruebas de estrés y descripción

Tipo de prueba	Descripción
Secuencial	Consiste en enviar una petición a la vez de manera secuencial al servidor sin esperar una respuesta exitosa o de rechazo
Concurrente	Consiste en enviar todas las peticiones al mismo tiempo al servidor sin esperar una respuesta exitosa o de rechazo
Lotes (batches)	Consiste en enviar un grupo de peticiones (diez peticiones) cada seis segundos, debido a que el tiempo de bloques de Polkadot es de seis segundos (Polkadot, 2024b), sin esperar una respuesta exitosa o de rechazo

Para crear las cuentas, los nombres y apellidos tienen de longitud entre cinco y seis caracteres, respectivamente, pues es la longitud media de los nombres y apellidos en el Perú (Sociedad LR, 2022), mientras que los *e-mails* tienen alrededor de veintidós caracteres (AtData, 2021).

Por otro lado, se realizaron pruebas de estrés con cien HME distribuidos aleatoriamente en los dos hospitales pertenecientes a HF, lo que garantizó la interoperabilidad entre ambas entidades. Esta prueba de estrés consistió en la generación de los cien HME, incluyendo los datos textuales en CouchDB y las imágenes convertidas a NFT en IPFS.

RESULTADOS

En la Tabla 5, se presentan las métricas obtenidas para las funcionalidades add_user, get_role y has_permissions, tras las pruebas de estrés con mil peticiones. En GET / has_permission/, la latencia media fue de 3,65 ms y la mediana de 3,51 ms, prácticamente con un consumo de CPU y RAM nulo. Esto confirma que las consultas de solo lectura al contrato son muy ligeras y pueden ejecutarse con alta frecuencia sin afectar los recursos del sistema.

La ruta GET /role/ mostró resultados similares, con una latencia media de 3,88 ms y mediana de 3,76 ms, manteniendo un uso de CPU mínimo y un incremento de memoria insignificante (0,0001 %) (Tabla 5). Ambas operaciones reflejan la eficiencia de las consultas de lectura en la red.

En cuanto a las operaciones de escritura, POST /create_user_with_dynamic_gas presentó una latencia media de 309,41 ms y mediana de 318,89 ms en modo secuencial, con un aumento casi nulo en el uso de CPU y solo un 0,0007 % de incremento en RAM (Tabla 5). El retardo provino principalmente del envío de la transacción a la red de Polkadot y del cálculo del tip dinámico previo a la firma, una sobrecarga razonable que mejora la fiabilidad sin penalizar el rendimiento local.

En modo *batch* (envío agrupado en lotes), la latencia media se elevó ligeramente a 365,66 ms (mediana de 364,55 ms) y el consumo de RAM aumentó hasta 0,009 %. Incluso así mantuvo un rendimiento eficiente (Tabla 5). Esto indica que el envío agrupado introduce una mínima sobrecarga, pero sigue siendo una opción viable para procesar múltiples transacciones de forma controlada y estable.

Tabla 5

Métricas estadísticas con mil peticiones

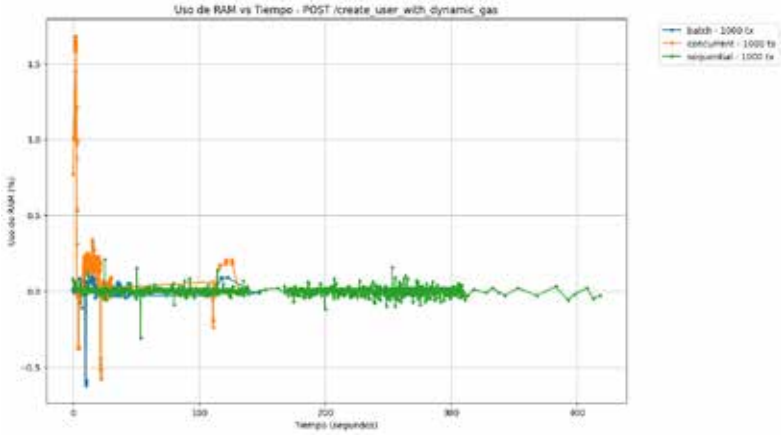
Ruta	Latencia media (ms)	Latencia mediana (ms)	Promedio de incremento del CPU (%)	Promedio de incremento de RAM (%)
GET /has_permission (secuencial)	3,65	3,51	0	0,0000
GET /role/ (secuencial)	3,88	3,76	0	0,0001
POST /create_user_with_dynamic_gas (secuencial)	309,41	318,89	0	0,0007
POST /create_user_with_dynamic_gas (batch)	365,66	364,55	0	0,0090

Como se observa en la Figura 12, la curva de uso de RAM muestra cómo, bajo carga secuencial (mil transacciones), disminuye y baja el uso del recurso, se mantiene uniforme a lo largo del tiempo con valores que oscilan entre 0,0 % a 0,3 % de uso. Con respecto a la modalidad concurrente de mil peticiones simultáneas, se observa que llega hasta 1,7 % para luego bajar rápidamente a 0 y tener una ligera subida posterior. Este comportamiento ocurre porque se colocan en cola las solicitudes y se van procesando lo más rápido posible.

Finalmente, con respecto al modo *batch*, este ofrece un comportamiento similar al modo secuencial en el uso de RAM, pero finaliza las peticiones más de cuatrocientos segundos antes. Los valores negativos en la gráfica aparecen porque, independientemente del uso de la arquitectura, el sistema operativo ejecuta procesos en segundo plano que se desarrollan en paralelo.

Figura 12

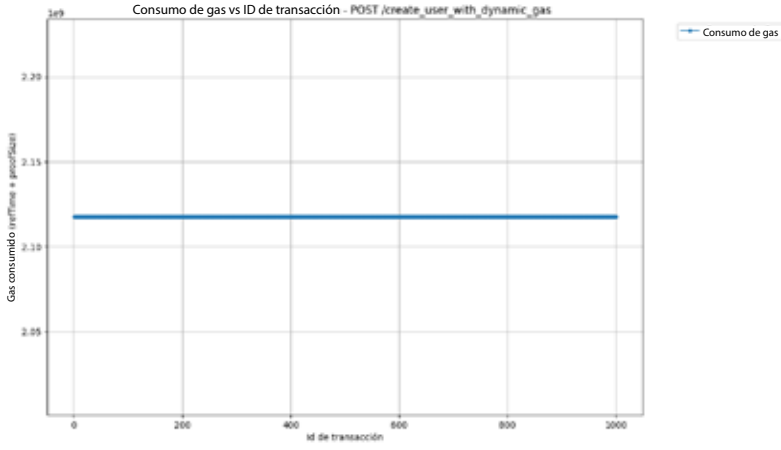
Uso de RAM en mil peticiones en el tiempo



En relación con la Figura 13, al graficar el *gas used* contra el identificador de cada transacción, vemos una línea prácticamente horizontal en torno a 2 117 705 426 unidades de gas. En cada transacción de las mil, la variabilidad es prácticamente nula, lo que implica que el mecanismo de gas dinámico consume una cantidad de gas similar en cada llamada. Esto se debe a que el peso de la transacción es el mismo para todas (misma longitud de nombre, apellido y *e-mail*) y que el tip aleatorio añadido sirve para priorizar las transacciones que llegan, sin afectar el costo base de gas. Cabe resaltar que el costo de transacción en la BPUBPER es fundamental para incentivar a los nodos de mantener la red funcionando.

Figura 13

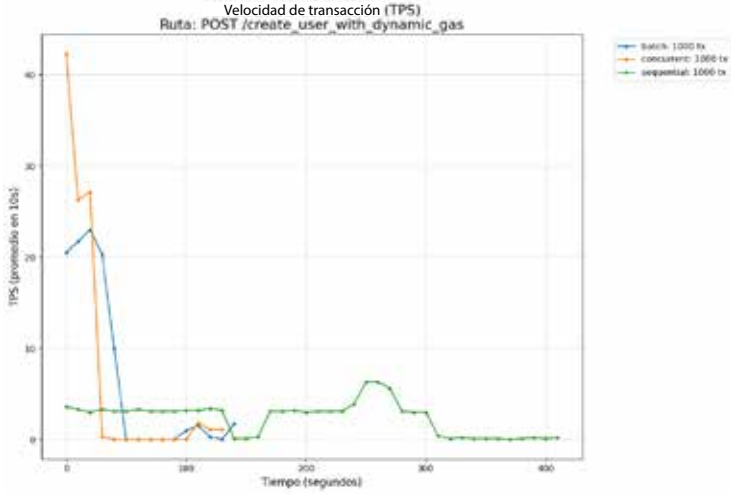
Uso de recursos consumo de gas vs ID de transacción



Con respecto a la Figura 14, el histograma de velocidad de transacción sobre tiempo muestra una acumulación muy marcada en el primer intervalo de cuarenta segundos. El pico inicial, fruto de inyectar las consultas en modo *batch* y el concurrente, significa que la *blockchain* ha recibido una gran cantidad de transacciones que van a ser procesadas por la *blockchain*. Asimismo, a pesar del modo concurrente de brindar todas las transacciones al inicio, tuvo una conclusión de envío de sus transacciones —al terminar de enviar las mil transacciones— casi igual al modo *batch*. Esto implica que no necesariamente hay una mejora significativa en la velocidad en envío de manera concurrente frente al envío en lotes. Con respecto al modo secuencial, este tuvo transacciones por segundo más estables que en los modos previos y una duración de más del doble de envío de transacciones.

Figura 14

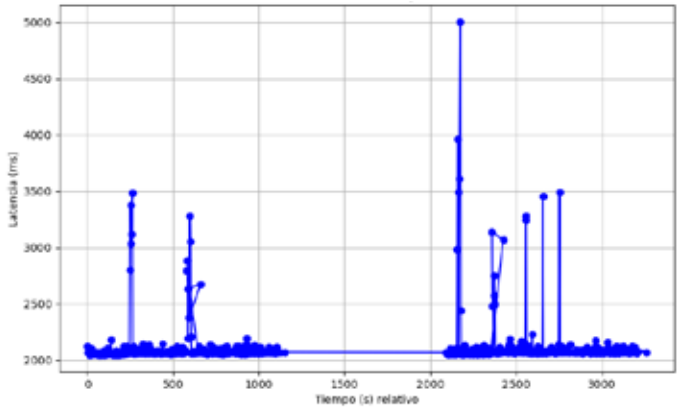
Velocidad de transacción sobre el tiempo



Por otro lado, se realizó la prueba de estrés en la BPRIV para la generación de cien HME, considerando las mismas métricas (latencia, uso de CPU y uso de RAM).

Figura 15

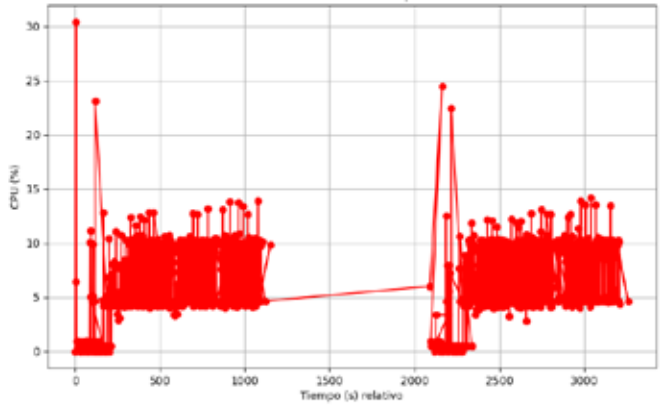
Latencia de consultas en milisegundos vs tiempo para cien creaciones de historiales médicos electrónicos



Como se puede visualizar en la Figura 16, con respecto a la latencia media de cada transacción, esta se sitúa muy cerca de los 2000 ms, lo que refleja el tiempo requerido para la propuesta, validación y *commit* en HF. Sin embargo, se aprecian brotes puntuales que llegan hasta los 3000 a 5000 ms, concentrados al inicio de cada gran bloque de transacciones (tanto en la primera fase como en el *minting*). Estos *outliers* coinciden con los momentos de alta concurrencia de llamadas (envío masivo de IPFS o de *minting* de NFT) y muestran cómo, ante picos de carga, el *ordering service* y los *peers* tardan más en procesar y confirmar los bloques. Aun así, la dispersión de la nube principal de puntos alrededor de los 2000 ms confirma un rendimiento bastante estable bajo condiciones normales.

Figura 16

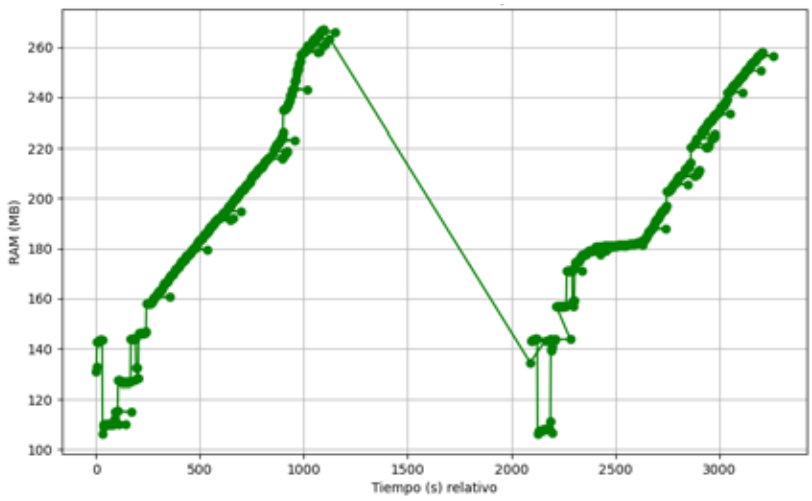
Porcentaje de uso de CPU vs tiempo para cien creaciones de historiales médicos electrónicos



Como podemos observar en la Figura 17, el gráfico de CPU revela dos fases bien diferenciadas a lo largo del *script*. En el arranque (primeros 200 segundos), el consumo oscila muy bajo, entre 0 % y 5 %, con picos aislados que alcanzan el 30 % al matricular al administrador y al procesar las primeras pocas transacciones. A continuación, durante la fase intensiva de creación de HME y subida de imágenes a IPFS (aproximadamente de 200 a 1000 segundos), el consumo se estabiliza entre un 6 % y un 14 %, lo que marca la mayor parte del trabajo de contrato más operaciones IPFS. Tras una pausa en la que el proceso se reinicia para la parte del *minting*, vuelve a aparecer un patrón muy similar (6 % a 12 %) con nuevos picos al iniciar esta segunda fase. En conjunto, se muestra que la mayor carga de CPU viene de los *loops* de invocación de Cintel y de *input/output* con IPFS, pero que la infraestructura de HF mantiene un uso moderado, incluso con cientos de transacciones.

Figura 17

Uso de RAM en megabytes vs tiempo para cien creaciones de historiales médicos electrónicos



Como podemos deducir de la Figura 17, el consumo de memoria arranca en unos 105 MB y crece de manera casi lineal hasta rondar los 265 MB conforme avanza la fase de HME más subida a IPFS (entre 0 y 1100 s). Después del corte (cuando se pasa al *minting*), observamos una caída brusca hasta ~135 MB, seguida de un nuevo crecimiento (hasta 260 MB en la segunda fase). Este patrón de subida sostenida indica que muchos objetos (*gateways*, contratos, *buffers* de IPFS, estructuras de datos de métricas) permanecen en memoria mientras el bucle avanza. Para evitar este *memory bloat*, convendría reciclar o liberar explícitamente las conexiones y datos intermedios entre lotes, o bien volcar métricas y desconectar *gateways* más a menudo.

En la Tabla 6, se presentan los resultados estimados para las métricas clave obtenidas a través de la simulación del despliegue de la propuesta de tesis en una red de HF configurada con dos organizaciones. Cabe resaltar que las métricas incluyen latencia y utilización de recursos.

Tabla 6

Resultados promedios de la simulación de la blockchain *privada*

Métricas	Resultados promedio
Latencia (creación)	2100 ms
Utilización de CPU	7 %
Utilización de memoria	190 MB

Se usaron librerías como *matplotlib* y *pidusage* para poder extraer la latencia y utilización de recursos respectivamente. Como se puede apreciar en la Tabla 6, la latencia promedio de las funciones "Creación de HME" y "Compartir los HME" con un doctor de la organización demoraron aproximadamente 2100 ms en generar cada caso particular. Por otro lado, en la sección de utilización de recursos, al realizar un descarte de los procesos no relacionados a la arquitectura, se puede evidenciar una utilización de CPU del 7 %. Esta es una métrica adecuada para el despliegue de ambas *blockchains* y sus respectivos servidores. Además, la utilización de la memoria ha sido baja, cuyo promedio fue de 190 MB para todo el proceso y mostró un cese de crecimiento conforme el proceso duraba más.

DISCUSIÓN

Los experimentos evidencian que las consultas puramente lectoras (GET /has_permission, GET /role) en modo secuencial se resuelven entre 3,65 y 3,88 ms, con un uso de CPU e incremento de RAM imperceptible para ambos casos (< 0,01 %). Estas cifras sitúan nuestro enfoque por encima de soluciones basadas únicamente en BPUB, las cuales suelen registrar latencias superiores y un mayor consumo de recursos. En el caso de operaciones de escritura (POST /create_user_with_dynamic_gas), la introducción de lógica de gas dinámico añade una sobrecarga de 100 a 200 ms en promedio, pero mantiene la estabilidad de uso de CPU (< 0,5 %) y RAM (< 0,3 %), lo que indica que el cálculo de tip no penaliza la ejecución local. Bajo concurrencia intensa, se generan picos de latencia de hasta 5,8 s, atribuibles principalmente a colas en la red de nodos y al proceso de consenso distribuido en Polkadot, y no al procesamiento en el servidor, lo que sugiere que futuros trabajos podrían investigar técnicas de escalado horizontal o *batching* más sofisticado. En la BPRIV, la creación masiva de cien HME arrojó una latencia media de ≈2 s, con CPU estabilizada en torno al 7 % y un crecimiento de RAM hasta

265 MB en fase intensiva. La aparición de *memory bloat* en picos sucesivos indica la conveniencia de liberar o reciclar conexiones IPFS y *gateways* entre lotes.

En conjunto, los resultados confirman que la separación de roles (pública para permisos y privada para datos clínicos) ofrece un equilibrio óptimo entre seguridad, rendimiento y escalabilidad, si bien requiere ajustes de optimización en la capa de red y en la gestión de memoria.

CONCLUSIONES

En conclusión, nuestra propuesta de BHIB demuestra que es posible integrar de manera efectiva la trazabilidad e inmutabilidad de una red pública permisionada (Polkadot) con la privacidad y el control de una red privada (HF) para la gestión de HME. Los resultados experimentales confirman lecturas ultrarrápidas (3,65-3,88 ms) con consumo local prácticamente nulo y escrituras estables (200-500 ms para usuarios y ≈2 s para HME), incluso bajo cargas de hasta 150 TPS, lo que mantiene el uso de CPU por debajo del 7 % y de RAM por debajo del 0,3 %. Aunque se observaron picos de latencia en escenarios altamente concurrentes y un crecimiento de memoria durante la fase de HME, estas limitaciones apuntan a la necesidad de incorporar mecanismos de *batching* adaptativo, autoescalado de nodos y *pooling* de conexiones IPFS para optimizar aún más el rendimiento.

En conjunto, este trabajo valida no solo la viabilidad técnica y la eficiencia operativa de un sistema de HME basado en BHIB, sino también su potencial para el despliegue en entornos sanitarios distribuidos, proporcionando una base sólida para futuras investigaciones en *sharding*, contratos más ligeros y auditoría clínica.

Finalmente, a pesar de los resultados positivos, la implementación presenta limitaciones, como la necesidad de una infraestructura robusta para mantener el rendimiento y la seguridad. Además, estudios futuros podrían explorar la optimización de los Cintel. Se recomienda la creación de sesiones temporales que permitan eliminar los permisos de visualización luego de un periodo de tiempo.

REFERENCIAS

Abouelmehdi, K., Beni-Hessane, A., & Khaloufi, H. (2018). Big healthcare data: Preserving security and privacy. *Journal of Big Data*, *5*(1), 1. https://doi.org/10.1186/s40537-017-0110-7

Abunadi, I., & Kumar, R. (2021). BSF-EHR: Blockchain security framework for electronic health records of patients. *Sensors*, *21*(8), 2865. https://doi.org/10.3390/s21082865

Ali, A., Rahim, H. A., Ali, J., Pasha, M. F., Masud, M., Rehman, A. U., Chen, C., & Baz, M. (2021). A novel secure blockchain framework for accessing electronic health records using multiple certificate authority. *Applied Sciences*, *11*(21), 9999. https://doi.org/10.3390/app11219999

Ali, A., Rahim, H. A., Pasha, M. F., Dowsley, R., Masud, M., Ali, J., & Baz, M. (2021). Security, privacy, and reliability in digital healthcare systems using blockchain. *Electronics*, *10*(16), 2034. https://doi.org/10.3390/electronics10162034

Amanat, A., Rizwan, M., Maple, C., Zikria, Y. B., Almadhor, A., & Kim, S. W. (2022). Blockchain and cloud computing-based secure electronic healthcare records storage and sharing. *Frontiers in Public Health*, *10*. https://doi.org/10.3389/fpubh.2022.938707

Antwi, M., Adnane, A., Ahmad, F., Hussain, R., Habib Ur Rehman, M., & Kerrache, C. A. (2021). The case of HyperLedger Fabric as a blockchain solution for healthcare applications. *Blockchain: Research and Applications*, *2*(1), 100012. https://doi.org/10.1016/j.bcra.2021.100012

AtData. (2021, 22 de junio). *How long is the average email address?* https://atdata.com/blog/long-email-addresses/

Awad Abdellatif, A., Samara, L., Mohamed, A., Erbad, A., Chiasserini, C. F., Guizani, M., O'Connor, M. D., & Laughton, J. (2021). MEdge-Chain: Leveraging edge computing and blockchain for efficient medical data exchange. *IEEE Internet of Things Journal*, *8*(21), 15762-15775. https://doi.org/10.1109/JIOT.2021.3052910

Bae, S., Kyung, D., Ryu, J., Cho, E., Lee, G., Kweon, S., Oh, J., Ji, L., Chang, E. I., Kim, T., & Choi, E. (2023). EHRXQA: A multi-modal question answering dataset for electronic health records with chest X-ray images. En A. Oh, T. Naumann & A. Globerson (Eds.), *NIPS '23: Proceedings of the 37th International Conference on Neural Information Processing Systems* (pp. 3867-3880). Curran Associates. https://dl.acm.org/doi/10.5555/3666122.3666292

Chelladurai, U., & Pandian, S. (2022). A novel blockchain based electronic health record automation system for healthcare. *Journal of Ambient Intelligence and Humanized Computing*, *13*(1), 693-703. https://doi.org/10.1007/s12652-021-03163-3

Chelladurai, U., Pandian, S., & Ramasamy, K. (2021). A blockchain based patient centric electronic health record storage and integrity management for e-Health systems. *Health Policy and Technology*, *10*(4), 100513. https://doi.org/10.1016/j.hlpt.2021.100513

Dagher, G. G., Mohler, J., Milojkovic, M., & Marella, P. B. (2018). Ancile: Privacy-preserving framework for access control and interoperability of electronic health records

using blockchain technology. *Sustainable Cities and Society*, *39*, 283-297. https://doi.org/10.1016/j.scs.2018.02.014

Exceline, E., & Nagarajan, S. (2024). Flexible access control mechanism for cloud stored EHR using consortium blockchain. *International Journal of System Assurance Engineering and Management*, *15*, 503-518. https://doi.org/10.21203/rs.3.rs-397642/v1

Fatokun, T., Nag, A., & Sharma, S. (2021). Towards a blockchain assisted patient owned system for electronic health records. *Electronics*, *10*(5), 580. https://doi.org/10.3390/electronics10050580

Guo, H., Li, W., Nejad, M., & Shen, C.-C. (2019). Access control for electronic health records with hybrid blockchain-edge architecture. En *2019 IEEE International Conference on Blockchain (Blockchain)* (pp. 44-51). IEEE. https://doi.org/10.1109/Blockchain.2019.00015

Haas, S. Wohlgemuth, I. Echizen, S., Sonehara, N., & Müller, G. (2011). Aspects of privacy for electronic health records. *International Journal of Medical Informatics*, *80*(2), e26-e31. https://doi.org/10.1016/j.ijmedinf.2010.10.001

Haddad, A., Habaebi, M. H., Elsheikh, E. A. A., Islam, Md. R., Zabidi, S. A., & Suliman, F. E. M. (2024). E2EE enhanced patient-centric blockchain-based system for EHR management. *Plos One*, *19*(4), e0301371. https://doi.org/10.1371/journal.pone.0301371

Hashim, F., Shuaib, K., & Sallabi, F. (2022). Connected blockchain federations for sharing electronic health records. *Cryptography*, *6*(3), 47. https://doi.org/10.3390/cryptography6030047

Hossain Faruk, M. J., Shahriar, H., Valero, M., Sneha, S., Ahamed, S. I., & Rahman, M. (2021). Towards blockchain-based secure data management for remote patient monitoring. En *2021 IEEE International Conference on Digital Health* (pp. 299-308). IEEE. https://doi.org/10.1109/ICDH52753.2021.00054

Huang, H., Sun, X., Xiao, F., Zhu, P., & Wang, W. (2021). Blockchain-based eHealth system for auditable EHRs manipulation in cloud environments. *Journal of Parallel and Distributed Computing*, *148*, 46-57. https://doi.org/10.1016/j.jpdc.2020.10.002

Hussien, H., Yasin, S., Udzir, N., & Ninggal, M. (2021). Blockchain-based access control scheme for secure shared personal health records over decentralised storage. *Sensors*, *21*(7), 2462. https://doi.org/10.3390/s21072462

Ink. (s. f.). *Ink vs. CosmWasm*. https://use.ink/docs/v5/ink-vs-cosmwasm/

International Organization for Standardization. (2019). *ISO 13606-1:2019(en) Health informatics — Electronic health record communication — Part 1: Reference model*. https://www.iso.org/obp/ui/en/#iso:std:iso:13606:-1:ed-2:v1:en

Jayabalan, J., & Jeyanthi, N. (2022). Scalable blockchain model using off-chain IPFS storage for healthcare data security and privacy. *Journal of Parallel and Distributed Computing*, *164*, 152-167. https://doi.org/10.1016/j.jpdc.2022.03.009

Jin, H., Luo, Y., Li, P., & Mathew, J. (2019). A review of secure and privacy preserving medical data sharing. *IEEE Access*, *7*, 61656-61669. https://doi.org/10.1109/ACCESS.2019.2916503

Kang, P., Yang, W., & Zheng, J. (2022). Blockchain private file storage-sharing method based on IPFS. *Sensors*, *22*(14), 5100. https://doi.org/10.3390/s22145100

Kaur, J., Rani, R., & Kalra, N. (2023). Attribute-based access control scheme for secure storage and sharing of EHRs using blockchain and IPFS. *Cluster Computing*, *27*, 1047-1061. https://doi.org/10.1007/s10586-023-04038-2

Kumari, D., Singh, A., Sunil, G., Mishra, K., & Panda, S. (2024). HealthRec-chain: Patient-centric blockchain enabled IPFS for privacy preserving scalable health data. *Computer Networks*, *241*, 110223. https://doi.org/10.1016/j.comnet.2024.110223

Lee, J.-S., Chew, C.-J., Liu, J.-Y., Chen, Y.-C., & Tsai, K.-Y. (2022). Medical blockchain: Data sharing and privacy preserving of EHR based on smart contract. *Journal of Information Security and Applications*, *65*, 103117. https://doi.org/10.1016/j.jisa.2022.103117

Li, P., Zhou, D., Ma, H., & Lai, J. (2024). Flexible and secure access control for EHR sharing based on blockchain. *Journal of Systems Architecture*, *146*, 103033. https://doi.org/10.1016/j.sysarc.2023.103033

Liu, J., Jiang, W., Sun, R., Kashif, A., Dahman, M.i, Hua, Q., & Yu, K. (2023). Conditional anonymous remote healthcare data sharing over blockchain. *IEEE Journal of Biomedical and Health Informatics*, *27*(5), 2231-2242. https://doi.org/10.1109/JBHI.2022.3183397

Mandarino, V., Pappalardo, G., & Tramontana, E. (2024). A blockchain-based electronic health record (EHR) system for edge computing enhancing security and cost efficiency. *Computers*, *13*(6), 132. https://doi.org/10.3390/computers13060132

Mani, V., Manickam, P., Alotaibi, Y., Alghamdi, S., & Khalaf, O. I. (2021). Hyperledger healthchain: Patient-centric IPFS-based storage of health records. *Electronics*, *10*(23), 3003. https://doi.org/10.3390/electronics10233003

Mauricio, D., Llanos-Colchado, P. C., Cutipa-Salazar, L. S., Castañeda, P., Chuquimbalqui-Maslucán, R., Rojas-Mezarina, L., & Castillo-Sequera, J. L. (2024). Electronic health record interoperability system in Peru using blockchain. *International Journal of Online and Biomedical Engineering*, *20*(3), 136-153. https://doi.org/10.3991/ijoe.v20i03.44507

Miyachi, K., & Mackey, T. K. (2021). hOCBS: A privacy-preserving blockchain framework for healthcare data leveraging an on-chain and off-chain system design. *Information Processing & Management*, *58*(3), 102535. https://doi.org/10.1016/j.ipm.2021.102535

Mohey Eldin, A., Hossny, E., Wassif, K., & Omara, F. A. (2023). Federated blockchain system (FBS) for the healthcare industry. *Scientific Reports*, *13*(1), 2569. https://doi.org/10.1038/s41598-023-29813-4

Montañez-Valverde, R. A., Montenegro-Idrogo, J. J., & Vásquez-Alva, R. (2015). Pérdida de información en historias clínicas: más allá de la calidad en el registro. *Revista Médica de Chile*, *143*(6), 812. https://doi.org/10.4067/S0034-98872015000600017

Mulligan, E., & Braunack-Mayer, A. (2004). Why protect confidentiality in health records? A review of research evidence. *Australian Health Review*, *28*(1), 48-55. https://doi.org/10.1071/AH040048

Nhan, T., Upadhyay, K., & Poudel, K. (2024). Towards patient-centric healthcare: Leveraging blockchain for electronic health records. En S. Zaza & C. Riemenscheinder (Eds.), *SIGMIS-CPR '24: Proceedings of the 2024 Computers and People Research Conference* (pp. 1-8). https://doi.org/10.1145/3632634.3655883

Omar, A. A., Jamil, A. K., Khandakar, A., Uzzal, A. R., Bosri, R., Mansoor, N., & Rahman, M. S. (2021). A transparent and privacy-preserving healthcare platform with novel smart contract for smart cities. *IEEE Access*, *9*, 90738-90749. https://doi.org/10.1109/ACCESS.2021.3089601

Polkadot. (2024a). *Transaction fees*. https://wiki.polkadot.network/docs/learn-transaction-fees

Polkadot. (2024b). *Consensus*. https://wiki.polkadot.com/learn/learn-consensus/

Puneeth, R. P., & Parthasarathy, G. (2024). Blockchain-based framework for privacy preservation and securing EHR with patient-centric access control. *Acta Informatica Pragensia*, *13*(1), 1-23. https://doi.org/10.18267/j.aip.225

Rajput, A. R., Li, Q., & Ahvanooey, M. T. (2021). A blockchain-based secret-data sharing framework for personal health records in emergency condition. *Healthcare*, *9*(2), 206. https://doi.org/10.3390/healthcare9020206

Samala, A. D., & Rawas, S. (2024). Transforming healthcare data management: A blockchain-based cloud EHR system for enhanced security and interoperability. *International Journal of Online and Biomedical Engineering*, *20*(2), 46-60. https://doi.org/10.3991/ijoe.v20i02.45693

Selvarajan, S., & Mouratidis, H. (2023). Author correction: A quantum trust and consultative transaction-based blockchain cybersecurity model for healthcare systems. *Scientific Reports*, *13*, 9409. https://doi.org/10.1038/s41598-023-36573-8

Sociedad LR. (2022). Top de los nombres más populares de niños y niñas en Perú, según Reniec. *La República*. https://larepublica.pe/sociedad/2022/09/12/reniec-top-de-los-nombres-mas-populares-de-ninos-y-ninas-en-peru-dni-nombres-comunes-peru-2022

Shen, B., Guo, J., & Yang, Y. (2019). MedChain: Efficient healthcare data sharing via blockchain. *Applied Sciences*, *9*(6), 1207. https://doi.org/10.3390/app9061207

Singh, A. P., Pradhan, N. R., Luhach, A. K., Agnihotri, S., Jhanjhi, N. Z., Verma, S., Kavita, Ghosh, U., & Roy, D. S. (2021). A novel patient-centric architectural framework for blockchain-enabled healthcare applications. *IEEE Transactions on Industrial Informatics*, *17*(8), 5779-5789. https://doi.org/10.1109/TII.2020.3037889

Tanwar, N., & Thakur, J. (2023). Patient-centric soulbound NFT framework for electronic health record (EHR). *Journal of Engineering and Applied Science*, *70*(33). https://doi.org/10.1186/s44147-023-00205-9

Uddin, M., S. Memon, M., Memon, I., Ali, I., Memon, J., Abdelhaq, M., & Alsaqour, R. (2021). Hyperledger fabric blockchain: Secure and efficient solution for electronic health records. *Computers, Materials & Continua*, *68*(2), 2377-2397. https://doi.org/10.32604/cmc.2021.015354

Uppal, S., Kansekar, B., Mini, S., & Tosh, D. (2023). HealthDote: A blockchain-based model for continuous health monitoring using interplanetary file system. *Healthcare Analytics*, *3*, 100175. https://doi.org/10.1016/j.health.2023.100175

Wang, M., Guo, Y., Zhang, C., Wang, C., Huang, H., & Jia, X. (2021). MedShare: A privacy-preserving medical data sharing system by using blockchain. *IEEE Transactions on Services Computing*, *16*(1). https://doi.org/10.1109/TSC.2021.3114719

Wu, G., Wang, S., Ning, Z., & Zhu, B. (2022). Privacy-preserved electronic medical record exchanging and sharing: A blockchain-based smart healthcare system. *IEEE Journal of Biomedical and Health Informatics*, *26*(5), 1917-1927. https://doi.org/10.1109/JBHI.2021.3123643

Xiao, Y., Xu, B., Jiang, W., & Wu, Y. (2021). The health chain blockchain for electronic health records: Development study. *Journal of Medical Internet Research*, *23*(1), e13556. https://doi.org/10.2196/13556

Zhang, G., Yang, Z., & Liu, W. (2022). Blockchain-based privacy preserving e-health system for healthcare data in cloud. *Computer Networks*, *203*, 108586. https://doi.org/10.1016/j.comnet.2021.108586