UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL

CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN

Tema:

MODELO DE ARQUITECTURA DE CADENA DE BLOQUES (BLOCKCHAIN) QUE PERMITA MANTENER LA INTEGRIDAD Y PRIVACIDAD DE LA INFORMACIÓN APLICANDO CONTRATOS INTELIGENTES.

Trabajo de titulación modalidad Proyecto de Investigación, presentado previo a la obtención del título de Ingeniero en Tecnologías de la Información.

ÁREA: Redes

LÍNEA DE INVESTIGACIÓN: Seguridad informática

AUTOR: Pablo Adrián Pérez Gallegos

TUTOR: Ing. David Omar Guevara Aulestia, Mg.

Ambato – Ecuador

agosto - 2023

APROBACIÓN DEL TUTOR

En calidad de tutor del trabajo de titulación con el tema: MODELO DE ARQUITECTURA DE CADENA DE BLOQUES (BLOCKCHAIN) QUE PERMITA MANTENER LA INTEGRIDAD Y PRIVACIDAD DE LA INFORMACIÓN APLICANDO CONTRATOS INTELIGENTES, desarrollado bajo la modalidad Proyecto de Investigación por el señor Pablo Adrián Pérez Gallegos, estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 17 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.3 del instructivo del reglamento referido.

Ambato, agosto 2023.

Ing. David Omar Guevara Aulestia, Mg.

TUTOR

AUTORÍA

El presente trabajo de titulación titulado: MODELO DE ARQUITECTURA DE CADENA DE BLOQUES (BLOCKCHAIN) QUE PERMITA MANTENER LA INTEGRIDAD Y PRIVACIDAD DE LA INFORMACIÓN APLICANDO CONTRATOS INTELIGENTES, es absolutamente original, auténtico y personal y ha observado los preceptos establecidos en la Disposición General Quinta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, agosto 2023.

Pablo Adrián Pérez Gallegos

C.C. 1718020819

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato para que reproduzca total o parcialmente este trabajo de titulación dentro de las regulaciones legales e institucionales correspondientes. Además, cedo todos mis derechos de autor a favor de la institución con el propósito de su difusión pública, por lo tanto, autorizo su publicación en el repositorio virtual institucional como un documento disponible para la lectura y uso con fines académicos e investigativos de acuerdo con la Disposición General Cuarta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato.

Ambato, agosto 2023.

Pablo Adrián Pérez Gallegos

C.C. 1718020819

AUTOR

APROBACIÓN DEL TRIBUNAL DE GRADO

En calidad de par calificador del informe final del trabajo de titulación presentado por el señor Pablo Adrián Pérez Gallegos, estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado MODELO DE ARQUITECTURA DE CADENA DE BLOQUES (BLOCKCHAIN) QUE PERMITA MANTENER LA INTEGRIDAD Y PRIVACIDAD DE LA INFORMACIÓN APLICANDO CONTRATOS INTELIGENTES, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 19 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.4 del instructivo del reglamento referido. Para cuya constancia suscribimos, conjuntamente con la señora Presidente del Tribunal.

Ambato, agosto 2023.

Firmado electrónicamente por:
ELSA PILAR URRUTIA URRUTIA

Ing. Elsa Pilar Urrutia Urrutia, Mg.

PRESIDENTE DEL TRIBUNAL

Firmado electrónicamente por:
FRANKLIN OSWALDO MAYORGA MAYORGA

Ing. Franklin Oswaldo Mayorga Mayorga, Mg.

PROFESOR CALIFICADOR

Firmado electrónicamente por:
RUBEN EDUARDO NOGALES PORTERO

Ing. Rubén Eduardo Nogales Portero, Mg.

PROFESOR CALIFICADOR

DEDICATORIA

Dedico este proyecto con amor y gratitud a mi madre Sabine Gallegos, cuyo apoyo constante y amor incondicional han sido mi fuente de fortaleza. Sus palabras de aliento han sido la fuerza que me empuja hacia adelante y en sus brazos encontré mi refugio.

A mi padre Pablo Pérez, que ha tenido la voluntad de ayudarme a levantar cuando estaba a punto de rendirme. Tu ejemplo de determinación y perseverancia me ha inspirado desde el principio.

A mis queridos abuelos Mauricio Gallegos y Yolanda Vásquez, cuyo amor y sabiduría han dejado una huella indeleble en mi camino. Este logro es un tributo a su amor y sacrificio.

AGRADECIMIENTO

Quiero expresar mi profundo agradecimiento a toda mi familia que siempre estuvieron pendientes de mi y contribuyeron para la realización de este objetivo personal.

A mi amigo Luis Barragán y a su familia que supieron abrirme sus brazos para brindarme apoyo y ayuda cuando más lo necesitaba.

A mi hermano Eduardo Pérez, que me sirvió de motivación para superar desafíos y a aspirar a ser la mejor versión de mí, espero que en mí encuentre el mejor modelo a seguir.

A mis amigos Andrés M, Galo S, Pablo A y Jonathan E, quienes me supieron guiar y ayudar en todo lo que necesitaba durante mi tiempo de estudios, su compañía y risas hicieron que los desafíos sean más ligeros.

También quiero agradecer a toda la FISEI, la calidad de educación que he recibido por parte de los docentes ha dejado una impresión duradera en mi formación.

ÍNDICE GENERAL DE CONTENIDOS

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

RESUMEN EJECUTIVO

La tecnología Blockchain y los Contratos Inteligentes han revolucionado la forma en que se interactúa con la información y los sistemas descentralizados. Con el objetivo de mantener la integridad y privacidad de los datos en un mundo digital cada vez más interconectado.

El presente proyecto se enfoca en el desarrollo de un modelo de arquitectura en Cadena de Bloques (Blockchain) aplicando Contratos Inteligentes con el propósito de mantener la integridad y privacidad de la información en diferentes aplicaciones y sistemas. El propósito es diseñar una arquitectura sólida y segura que permita aprovechar las ventajas de la tecnología Blockchain y los Contratos Inteligentes para garantizar la inmutabilidad de los datos.

Este proyecto se presenta como una guía y referencia para desarrolladores, empresas e instituciones que buscan aprovechar la tecnología Blockchain y los Contratos Inteligentes de manera efectiva y segura. Al combinar la integridad de la Cadena de Bloques con las capacidades de automatización de los Contratos Inteligentes, se abre un abanico de posibilidades para mejorar la confiabilidad, transparencia y privacidad en diferentes procesos y sistemas.

Palabras clave: Blockchain, contratos inteligentes, ethereum, arquitectura.

ABSTRACT

Blockchain technology and Smart Contracts have revolutionized the way information and decentralized systems are interacted with. With the aim of maintaining data integrity and privacy in an increasingly interconnected digital world.

This project focuses on the development of an architecture model in Blockchain using Smart Contracts. This research seeks to design a robust and secure structure that harnesses the advantages of Blockchain technology and Smart Contracts to ensure data immutability.

This project is presented as a guide and reference for developers, companies, and institutions looking to effectively and securely leverage Blockchain technology and Smart Contracts. By combining the integrity of the Blockchain with the automation capabilities of Smart Contracts, a range of possibilities opens up to enhance reliability, transparency, and privacy in various processes and systems.

Key words: Blockchain, smart contracts, ethereum, architecture.

CAPÍTULO I.- MARCO TEÓRICO

1. Tema de Investigación

MODELO DE ARQUITECTURA DE CADENA DE BLOQUES (BLOCKCHAIN) QUE PERMITA MANTENER LA INTEGRIDAD Y PRIVACIDAD DE LA INFORMACIÓN APLICANDO CONTRATOS INTELIGENTES.

1.1.1 Planteamiento del Problema

En los últimos años, la tecnología Blockchain ha tenido un crecimiento exponencial, sobre todo en el ámbito de las criptodivisas. En este ámbito también han ido ganando protagonismo los denominados Smart Contracts o Contratos Inteligentes debido a su creciente versatilidad en el ámbito contractual, dado que esta tecnología permite automatizar la ejecución de una o varias obligaciones contractuales y es aplicable a una gama amplísima de supuestos. No obstante, esta tecnología está en una etapa de expansión [1].

En el Blockchain la base de datos descentralizada es invariable e igualmente accesible para cada individuo que entra en la red. Esto quiere decir que cualquier sistema transferido o creado a base del Blockchain, así como sus transacciones pueden ser visualizados. En América Latina la transparencia puede ser de beneficio para las dos partes: para los ciudadanos que controlan al Gobierno y para el Gobierno que inspecciona a los ciudadanos. Si las transacciones del Gobierno están registradas en una de las Cadenas de Bloques, quedará garantizado que nadie se apropiará ilícitamente del dinero del Estado [2].

El problema a nivel de América Latina radica en que la tecnología Blockchain y los Contratos Inteligentes pueden ser utilizados para almacenar información altamente sensible, como datos personales, financieros y de salud. Si esta información cae en manos equivocadas, puede tener consecuencias graves para los usuarios y las organizaciones que la utilizan.

Si bien se ha visto que en el Ecuador existe por una parte normativa Constitucional que protege a los ciudadanos con relación a su información y las bases de datos y además en la legislación secundaria existe normativa que reconoce el valor jurídico del contenido digital y de su soporte es importante definir si se legisla sobre la tecnología de Cadena de Bloques excluyendo las monedas virtuales o si se pretende reconocer las criptomonedas y el Blockchain de ellos depende que reformas deberían realizarse, en el primer caso es reforzar el tema de uso de datos y de los problemas de anonimato y los tremas de gestión pública y privada y en el segundo reconocer un nuevo sistema de pagos donde tendrían que apartarse todo el sistema normativo financiero y tributario [3].

En Ecuador, otro problema se relaciona con la falta de conocimiento y adopción de soluciones de privacidad de datos en la Cadena de Bloques. Muchas empresas y organizaciones aún no están familiarizadas con técnicas de privacidad de datos en la Cadena de Bloques, como la encriptación y los Contratos Inteligentes, y no las implementan en sus aplicaciones. Esto resulta en una vulnerabilidad de la información almacenada en la Cadena de Bloques ante posibles ataques y manipulaciones.

1.2 Antecedentes Investigativos

Revisando la investigación bibliográfica en algunas universidades del Ecuador y del exterior se han encontrado trabajos que servirán de apoyo en el trabajo de investigación:

Luis Eduardo Rosero Correa en su tesis [4] “Propuesta de una aplicación basada en la tecnología Blockchain para el registro de títulos académicos”, trabajo realizado como tesis de la Universidad Central del Ecuador, en el año de 2019 se obtiene que:

- Hablar de Blockchain es, sin lugar a duda hablar de una revolución tecnológica significativa dado que ofrece un potencial extraordinario especialmente en aquellas áreas en donde se requiere un registro confiable e inmutable de cada

transacción como por ejemplo el registro de títulos académicos, es por eso que su utilidad trasciende más allá de las criptomonedas y con los instrumentos adecuados como por ejemplo Ethereum, se puede explotar todo ese potencial.

- A pesar de que el primer enfoque de la Cadena de Bloques se centró en las criptomonedas y las grandes empresas brindan un fuerte apoyo a esta iniciativa, también existe un proceso bastante importante en la adopción de los Contratos Inteligentes que se puede evidenciar por proyectos como Ethereum, NEO, Taboow o Stratris y muchas otras que ofrecen una plataforma para Contratos Inteligentes.
- Al ser los Contratos Inteligentes junto con Blockchain tecnologías nuevas y en proceso de madurez, es difícil encontrar modelos de buenas prácticas referentes al desarrollo, al rendimiento o a la seguridad, que permitan encaminar el proceso de creación hacia la meta de conseguir un producto robusto y de calidad.

Bryan Andrés Castro Tapia en su tesis [5] "Propuesta de una aplicación basada en la tecnología Blockchain y Smart Contracts para el registro de contratos de arrendamiento.", trabajo realizado como tesis de la Universidad Central del Ecuador en el año de 2021 se determina que:

- La ejecución de transacciones en Blockchain tiene un costo relacionado con la utilización de recursos computacionales usados para poder compilar y añadir cada bloque de transacción a la red. Este valor se tiene que pagar utilizando la divisa de Ethereum lo que significa reemplazar el dólar por una criptomoneda para sustentar las transacciones, aunque no sea la moneda oficial. Esto crea una oportunidad de mejorar el comercio y la economía, eliminando el coste que representa mantener dinero físico, además de excluir intermediarios como bancos, jueces y abogados que solo incrementan el valor de las tarifas para la industria inmobiliaria.
- Blockchain es una tecnología tan disruptiva que está cambiando la forma de funcionamiento de muchas industrias, al agregar este enorme potencial de

generar confianza en las relaciones sin un agente central, se puede personalizar una implementación para cada empresa que requiera almacenar su información de forma segura fijando una mejora de sus procesos que ahora es posible gracias al mundo globalizado y totalmente conectado que se goza en esta época.

Julio Antonio Orrala Moreira en su tesis [6] “Diseño de una guía para la elaboración de smart contracts” trabajo realizado como tesis de la Universidad Estatal Península de Santa Elena, en el año 2022 se concluye que:

- Para el desarrollo de la guía se aplicó conocimiento de Blockchain y criptomonedas, dando a conocer una de las formas de realizar Contratos Inteligentes.
- Las fases aplicadas en el proyecto tienen su base en una propuesta metodológica para el desarrollo de Smart Contracts; la información referente al desarrollo se centró en otro tipo de moneda y no la plasmada en el proyecto, por ello se realiza el caso de estudio dentro de la red de Cardano debido a los beneficios que trae usar esta Cadena de Bloques.
- La recolección de información es crucial para detectar los datos e información que deberá ir plasmada en el contrato y ser subidos a la Blockchain, por ello es necesario conocer cómo funciona la empresa y sus procesos de negocio, tomando uno de estos procesos como base para la implementación del Contrato Inteligente. A través del uso del sistema web interno el gimnasio se mejora el desempeño de los procesos de registro de clientes, valoraciones físicas, el servicio médico cuenta con herramientas tecnológicas que facilitan su trabajo diario.

Luis Rodrigo Álvarez Rojas en su tesis [7] “Análisis de la tecnología Blockchain, su entorno y su impacto en modelos de negocios”, trabajo de tesis realizada para la Universidad Técnica Federico Santa María de Chile, en el año 2018 concluye que:

- Actualmente la tecnología Blockchain se encuentra en etapa de crecimiento apoyado por grandes empresas que apuestan por su desarrollo y potenciales usos. Existen una variedad de industrias en el ámbito privado que ven a la tecnología Blockchain como una forma de mejorar sus procesos reduciendo tiempos de procesos, reducción de costos, aumento de la seguridad, transparencia y estabilidad del sistema. Sin embargo, el verdadero potencial de la tecnología se puede apreciar en una red mayor mediante una rápida adopción, con la generación de nuevos modelos de negocio mediante aplicaciones públicas descentralizadas apoyando el desarrollo de la infraestructura en un aspecto económico, empresarial y social.

- El análisis de la tecnología Blockchain requiere de la comprensión total de su ecosistema, entendiendo el aporte generado a través de la cadena de valor de la industria por cada uno de los actores de acuerdo a sus modelos de negocios. Esto permite evaluar posibilidades de generación de nuevos modelos de negocios, reestructuración de los existentes para las empresas que serán impactadas y la creación de nuevos negocios que utilicen el desarrollo actual de la tecnología.

- Si bien en la actualidad la tecnología Blockchain está siendo adoptada principalmente por la industria financiera, para la mejora de sus procesos y sus servicios, también es factible su utilización en otros sectores del ámbito privado como cadena de suministro, legal, salud, social, educación, energía y sustentabilidad.

Jenny Alexandra Triana Casallas en su tesis doctoral [8] “Meta-Modelo de Contratos Inteligentes usando cadenas de bloques aplicado al sector público”, trabajo realizado para la Universidad Oviedo de Colombia en el año 2021 se determina que:

- Con el correcto diseño y aplicación de un sistema que se beneficie del uso de Blockchain público con Smart Contracts, que sea fiable, inalterable y seguro; se

pueden mejorar los índices de transparencia de la administración pública colombiana en favor de la disminución de la corrupción, toda vez que, si alguien realiza un cambio, se verá como las partidas presupuestales cambian, debido a que con Blockchain los registros son inalterables y con la implementación de Smart Contracts, las etapas de contratación y de ejecución presupuestal en las que hay mayor susceptibilidad de materialización de riesgos de corrupción, serán autoejecutables una vez se cumplan los acuerdos o reglas establecidos previamente por las partes, de manera que no hay intervención de terceros que validen el cumplimiento de las condiciones contractuales, por lo que cada modificación será evidente y transparente.

- Por su parte, la solución propuesta presenta un meta-modelo y un prototipo de integración de Blockchain público con Smart Contracts que pretenden contribuir al desarrollo de aplicaciones para disminuir la corrupción y mejorar los índices de transparencia en la contratación en el sector público.

1.3 Fundamentación Teórica

Seguridad Informática

La Seguridad Informática es la disciplina que, con base en políticas y normas internas y externas de la empresa, se encarga de proteger la integridad y privacidad de la información que se encuentra almacenada en un sistema informático, contra cualquier tipo de amenazas, minimizando los riesgos tanto físicos como lógicos, a los que está expuesta. Esta definición se puede complementar señalando que en caso de que una amenaza a la seguridad se haga efectiva, debe procurar recuperar la información dañada o robada [9].

Lo primero que se debe mencionar es que en muchos casos se suelen confundir dos conceptos la seguridad informática y la seguridad de la información, aunque suenen muy parecidos tienen puntos clave que hacen una diferencia. La seguridad informática se encarga de la seguridad del medio informático, según varios autores la informática es

la ciencia encargada de los procesos, técnicas y métodos que buscan procesar almacenar y transmitir la información, mientras tanto la seguridad de la información no se preocupa sólo por el medio informático, se preocupa por todo aquello que pueda contener información, en resumen, esto quiere decir que se preocupa por casi todo, lo que conlleva a afirmar que existen varias diferencias, pero lo más relevante es el universo que manejan cada uno de los conceptos en el medio informático [10].

Se puede definir a la Seguridad Informática como la disciplina encargada de plantear y diseñar las normas, procedimientos, métodos y técnicas con el fin de obtener que un sistema de información sea seguro, confiable y sobre todo que tenga disponibilidad [10].

Ciberseguridad

La Ciberseguridad aprovecha el potencial de herramientas, métodos y recursos para prevenir o resistir eventos dañinos de seguridad. Se trata de una práctica o disciplina enfocada en reducir el riesgo cibernético a través de la protección de toda la infraestructura de tecnologías de la información, incluyendo los sistemas, las aplicaciones, el hardware, el software y los datos.

La Ciberseguridad es una práctica integral que abarca tecnologías, procesos y métodos para defender los sistemas informáticos, los datos y las redes. Para entender mejor qué es la ciberseguridad, se va a dar a conocer todas las áreas que se pueden y deben proteger [11]:

- Gestión de identidad y seguridad de datos: Este segmento se enfoca en los procesos que permiten la autorización y autenticación de usuarios en los sistemas de información empresariales. Involucra el uso de los protocolos de autenticación para controlar el acceso a los sistemas y datos.

- Seguridad de las aplicaciones: Abarca la implementación de diferentes defensas en el software y los servicios empresariales. Implica el diseño de arquitecturas de aplicaciones seguras para minimizar la posibilidad de acceso no autorizado.

- Seguridad móvil: Es el área de ciberseguridad encargada de proteger la información personal y empresarial tanto almacenada en dispositivos móviles como aquella a la que se accede a través de los mismos.
- Seguridad en la nube: Se refiere a la creación de arquitecturas y aplicaciones seguras hosteadas en la nube, para empresas que utilizan proveedores de servicios de nube.

Seguridad de la red: A través de mecanismos de hardware y software, protege la red y la infraestructura, de accesos no autorizados y otros abusos.

Privacidad de la Información

En general, la privacidad de los datos o Privacidad de la Información significa la capacidad de una persona para determinar por sí misma cuándo, cómo y hasta qué punto se comparte o se comunica a otros su información personal. Esta información personal puede ser el nombre, la ubicación, la información de contacto o el comportamiento en línea o en el mundo real. Al igual que alguien puede querer excluir a personas de una conversación privada, muchos usuarios de Internet quieren controlar o evitar que se recopilen ciertos tipos de datos personales. [12]

A medida que el uso de Internet ha ido aumentando a lo largo de los años, también lo ha hecho la importancia que tiene la privacidad de los datos. Los sitios web, las aplicaciones y las plataformas de las redes sociales a menudo necesitan recopilar y almacenar datos personales de los usuarios para poder prestar sus servicios. Sin embargo, algunas aplicaciones y plataformas pueden exceder las expectativas de los usuarios en lo que respecta a la recopilación y uso de datos, afectando a la privacidad de los usuarios mucho más de lo esperado. Puede que otras aplicaciones y plataformas no pongan las protecciones adecuadas en torno a los datos que recogen, lo que puede dar lugar a una fuga de datos que ponga en peligro la privacidad del usuario [12].

Cadena de Bloques (Blockchain)

Aunque originalmente la Cadena de Bloques fue creada para almacenar el historial de transacciones del bitcoin, con el paso del tiempo se le ha visto gran potencial para ser aplicada en otros ámbitos debido a las propiedades que ofrece. La Blockchain proporciona una base de datos distribuida inmutable basada en una secuencia creciente de bloques. Estos bloques, al ser públicos, conforman un sistema abierto que potencia la confianza en base a la transparencia y a la solidez de la técnica de construcción de la Blockchain. El sistema, aunque es abierto, es también semi-anónimo: los usuarios se identifican con claves públicas (pseudónimos), no con sus identidades reales. En este contexto, se puede encontrar una primera relación entre la Blockchain y big data: la necesidad de asegurar un entorno de pagos legal y libre de fraudes ha llevado al desarrollo de herramientas de análisis basadas en técnicas de big data para procesar la gran cantidad de datos representados en la Blockchain (Ron, 2013, April) y (Reid, 2013). Por tanto, el anterior, es un posible caso de uso de big data para mejorar los procesos de inserción de datos en la Blockchain [13].

Registro Distribuido

El Registro Distribuido es un sistema que permite el registro de transacciones y la gestión de datos simples y complejos. Este sistema está vinculado a una red de bases de datos no unívoca, donde los datos son replicados de manera continua y asíncrona entre todos los usuarios que utilizan dicho sistema. La ubicación del registro se encuentra descentralizada entre los diferentes usuarios involucrados en la red, quienes mantienen copias descentralizadas en sus bases de datos individuales. El objetivo principal del Registro Distribuido es garantizar la seguridad de las transacciones mediante el uso de criptografía, lo que asegura su inmutabilidad, verificabilidad, auditabilidad y consistencia temporal [14].

Contratos Inteligentes (Smart Contracts)

Un Contrato Inteligente posee la capacidad de ejecutarse y hacer cumplir sus términos de manera autónoma y automática, sin la intervención de intermediarios o mediadores. Al no estar expresado en lenguaje verbal o escrito, evita la ambigüedad en su

interpretación. Los Contratos Inteligentes son esencialmente "scripts" o códigos informáticos escritos en lenguajes de programación. Esto significa que los términos del contrato se expresan mediante sentencias y comandos en el código que lo compone.

La validez de un Contrato Inteligente no depende de autoridades, ya que su naturaleza se basa en la visibilidad de su código para todos los involucrados y en la incapacidad de alterarlo debido a su existencia en la tecnología Blockchain. Esto le otorga un carácter descentralizado, inmutable y transparente [15].

Redes P2P (Peer-to-Peer)

Una red P2P, cuyas siglas significa red de pares o red entre iguales en inglés, es una red en la que los nodos cumplen la función de servidores y de clientes al mismo tiempo, sin que exista ningún tipo de jerarquía al respecto. Así, en una red de estas características cada ordenador o dispositivo estaría en un plano de igualdad con los demás, provocando la existencia de una comunicación de tipo horizontal. Esto permite el intercambio directo de información, en cualquier formato entre los terminales interconectados. Este modelo de red contrasta con el clásico modelo cliente-servidor, el cual se rige mediante una estructura donde no hay ningún tipo de distribución de tareas entre sí, solo una comunicación entre terminal y usuario, por lo que estos no pueden intercambiar roles [16].

1.4 Objetivos

1.4.1 Objetivo General

- Desarrollar un modelo de arquitectura de Cadena de Bloques (Blockchain) que permita mantener la integridad y privacidad de la información aplicando Contratos Inteligentes.

1.4.2 Objetivos Específicos

- Identificar los riesgos asociados con la privacidad de los datos en la Cadena de Bloques (Blockchain).

- Evaluar la capacidad de los Contratos Inteligentes, sus funciones y características que permiten el tratamiento de datos de manera eficiente y segura.
- Diseñar el modelo de arquitectura de Cadena de Bloques (Blockchain) aplicando Contratos Inteligentes.

CAPÍTULO II.- METODOLOGÍA

2.1 Materiales

Para la recolección de información del presente proyecto de investigación se utilizaron fichas bibliográficas puesto que estas permitieron organizar y estructurar la información clave de una manera clara y concisa, para así obtener una idea más definida de la investigación a realizarse y poder sustentarla.

Tabla 1 Formato Ficha Bibliográfica

FICHA BIBLIOGRÁFICA	
TEMA	
TESIS	
PROPÓSITO	
IDEAS CENTRALES	
CONCEPTOS CLAVES	
CONCLUSIONES	
APORTE	
AÑO DE PUBLICACIÓN	

Elaborado por: El investigador

2.2 Métodos

2.2.1 Modalidad de la Investigación

La investigación se considera bibliográfica puesto que se apoyará en libros, revistas, trabajos de titulación del área informática, documentos técnicos y artículos que aporten a la investigación.

2.2.2 Población y Muestra

El proyecto al ser de tipo bibliográfico no se utilizará población y muestra.

2.2.3 Recolección de Información

Las fichas bibliográficas que se muestran a continuación presentan la información con relevancia a la investigación con respecto al tema y a sus variables, con el propósito de cumplir los objetivos.

Tabla 2 Ficha Bibliográfica 1

FICHA BIBLIOGRÁFICA	
TEMA	La seguridad y privacidad del Blockchain, más allá de la tecnología y las criptomonedas.
TESIS	El artículo ofrece un recorrido sobre los retos en la seguridad y la regulación de la protección de datos de carácter personal que plantea la tecnología Blockchain.
PROPÓSITO	Dar a conocer el potencial de Blockchain y también cómo funciona dicha tecnología en el aspecto de seguridad y privacidad.

IDEAS CENTRALES	• El Blockchain es una tecnología conceptualmente segura gracias a su naturaleza distribuida, la irreversibilidad de las transacciones y el uso intensivo de cifrado. Las vulnerabilidades surgen habitualmente como resultado de la implementación de las plataformas y aplicaciones, es decir, se vinculan al desarrollo del código informático, de los protocolos de comunicación o de la simplificación de los mecanismos de validación y consenso de los bloques [17]. • Además de errores de programación, las tecnologías Blockchain se enfrentan a riesgos que tienen que ver con las técnicas criptográficas que aseguran la confidencialidad y la integridad del registro de las transacciones [17].
CONCEPTOS CLAVES	Contiene conceptos de privacidad, seguridad, Reglamento General de Protección de Datos, Smart Contracts.
CONCLUSIONES	El Blockchain es una tecnología conceptualmente muy segura que se ve expuesta en el curso de su implementación a errores y vulnerabilidades propios de cualquier sistema de información, añadidos a los específicos de esta tecnología. A esto se suman los retos de seguridad, interoperabilidad y tecnológicos derivados de su progresiva madurez, complejidad, falta de estandarización y diversidad de protocolos [17].

APORTE	Brinda información de seguridad y privacidad, así también sobre las vulnerabilidades y desafíos de Blockchain.
AÑO DE PUBLICACIÓN	2019

Elaborado por: El investigador

Tabla 3 Ficha Bibliográfica 2

FICHA BIBLIOGRÁFICA	
TEMA	Riesgos para la seguridad de la innovadora plataforma “Blockchain”.
TESIS	El tema trabajado trata de intentar comprender algunos desafíos de la tecnología a partir de la innovación actual.
PROPÓSITO	Identificar algunos riesgos que se deben tener en cuenta en la tecnología Blockchain.

IDEAS CENTRALES	• Según Thiago Marques, experto digital de Kaspersky Labs, las formas más utilizadas por los delincuentes para obtener las contraseñas son: el phishing, programa maligno, infección a través de servidores de grandes organizaciones y JavaScript en sitios infectados [18]. • Como antecedentes de las amenazas latentes contra la tecnología Blockchain desde sus inicios en 2008, las plataformas han enfrentado diversas amenazas que ponen en peligro su estabilidad, un claro ejemplo es el mencionado por la firma DELOITTE en su artículo titulado Blockchain & ciberseguridad, donde se menciona la docilidad de las transacciones, con un error que se detectó cuando las transacciones se encontraban en un estado de validación pendiente, dio como resultado un ataque a la red Bitcoin en 2014 y posteriormente en 2016, un hacker aprovechó los contratos inteligentes en Ethereum, y la forma en que se pueden utilizar, para crear un desbordamiento en la red, hasta el punto en que la creación de bloques y la validación de las transacciones se vieron gravemente afectadas [18].
CONCEPTOS CLAVES	Contiene conceptos como phishing, malware, infección a través de servidores, JavaScript en sitios infectados.

CONCLUSIONES	• Las nuevas tecnologías traen de la mano nuevos problemas para la seguridad. Lo que hoy es seguro mañana no lo será, esto debido al ingenio de los seres humanos y así como hay personas creativas quienes buscan crear nuevas oportunidades para la sociedad, también existen quienes buscan atacarlas con el fin de obtener beneficio económico sin mayor esfuerzo [18]. • Finalmente, aunque existan riesgos que generen resistencia a la implementación de nuevas tecnologías, esta es una herramienta que puede aumentar la capacidad de las personas y las organizaciones y ayudar a resolver problemas existentes, además la tecnología se encuentra día a día en un crecimiento acelerado, por lo que es importante mantenerse actualizados, lo que incluye tener un conocimiento previo de lo que se va a utilizar [18].
APORTE	Este trabajo de grado aporta información de algunos de los riesgos a tomar en cuenta en la tecnología Blockchain.
AÑO DE PUBLICACIÓN	2019

Elaborado por: El investigador

Tabla 3 Ficha Bibliográfica 3

FICHA BIBLIOGRÁFICA	
TEMA	Blockchain: la revolución industrial de internet.
TESIS	Libro que presenta información detallada de las diferentes características, funciones, aplicaciones de Blockchain para un entendimiento mas claro de esta tecnología.
PROPÓSITO	Profundizar sobre la tecnología Blockchain dando a conocer su origen, potencial, sus aplicaciones, seguridad, etc.

IDEAS CENTRALES	Cada vez aparecen nuevas técnicas que intentan que los sistemas revelen información. Estas amenazas están también presentes en las cadenas de bloques. He aquí algunas de ellas [19]: • Ingeniería social. • Servicios en la nube. • Bring Your Own Device (BYOD). • Factores de riesgo interno. Tradicionalmente, las medidas de seguridad persiguen preservar una serie de propiedades sobre el sistema en el que se aplican, propiedades que son válidas no sólo para las soluciones que se construyen sobre una cadena de bloques, sino también para la propia cadena de bloques. Las que se explican a continuación son básicas [19]: • Confidencialidad. • Integridad de la información. • Autenticación de usuario. • Autenticación de remitente y destinatario. • No repudio en origen y destino.
CONCEPTOS CLAVES	Brinda conceptos de confidencialidad, integridad de la información, autenticación de usuario, ataques clásicos a Blockchain.
CONCLUSIONES	Blockchain siendo una tecnología revolucionaria no está exenta de amenazas o vulnerabilidades como se puede observar en la recopilación de antecedentes como los ataques clásicos a Blockchain que no solo afecta a las criptomonedas sino en sí a esta tecnología. Si bien no solo son los únicos modelos de ataques que existen, pueden servir como aviso.

APORTE	Este libro aporta bastante información sobre la seguridad en Blockchain, así como antecedentes de vulnerabilidades dentro de esta tecnología.
AÑO DE PUBLICACIÓN	2017

Elaborado por: El investigador

Tabla 4 Ficha Bibliográfica 4

FICHA BIBLIOGRÁFICA	
TEMA	Seguridad en la Blockchain de Ethereum: explotación y mitigación de vulnerabilidades modernas en Smart Contracts.
TESIS	El artículo presenta un recorrido desde los inicios de la tecnología Blockchain describiendo a profundidad su funcionamiento y diferentes mecanismos que lo componen.
PROPÓSITO	Buscar las vulnerabilidades mas comunes en los Smart Contracts de la tecnología Ethereum.

IDEAS CENTRALES	• La seguridad en los Smart Contracts es una de las consideraciones que se debe tener en cuenta a la hora de desarrollar cualquier aplicación distribuida basada en esta tecnología [20]. • Los Smart Contracts ejecutan el código exactamente como ha sido escrito, que no es siempre lo que el desarrollador espera que sea, y teniendo en cuenta que estos contratos son públicos y que cualquier usuario de la red puede interactuar con ellos [20].
CONCEPTOS CLAVES	Contiene conceptos como escalabilidad, robustez, descentralización, seguridad, redes P2P.
CONCLUSIONES	Los Smart Contracts no admiten modificaciones en su código una vez desplegados, lo que implica que cualquier error en su código es irrevocable en la mayoría de los casos, ya que esta tecnología impide cualquier tipo de parcheo [20]. Además, cada uno de estos errores probablemente cause pérdidas económicas, por lo tanto, es importante escribir código de calidad desde las etapas más tempranas de su desarrollo, aplicando patrones de desarrollo seguro desde el comienzo [20].
APORTE	Brinda información sobre las vulnerabilidades de los Smart Contracts.
AÑO DE PUBLICACIÓN	2022

Elaborado por: El investigador

Tabla 5 Ficha Bibliográfica 5

FICHA BIBLIOGRÁFICA	
TEMA	Identificación y control de riesgos en procesos validados con Blockchain.
TESIS	El artículo ofrece una investigación detallada y extensa de los riesgos que se puede tener dentro de la tecnología Blockchain.
PROPÓSITO	Identificar los riesgos que se pueden dar en procesos con Blockchain y como se los podría controlar.
IDEAS CENTRALES	• Esta tecnología ha nacido con una aureola de seguridad que rápidamente ha sido aceptada por la industria sin conocer en profundidad cómo funciona realmente o cómo se integran las aplicaciones que hacen uso de ella, lo que supone un riesgo [21]. • Si bien esta tecnología presenta ventajas como el manejo de la confidencialidad o la integra, incurre en diversas vulnerabilidades de seguridad, algunas de ellas intrínsecas a la propia naturaleza de la Blockchain o a las implementaciones que se han realizado [21].
CONCEPTOS CLAVES	Contiene conceptos de principales amenazas y sus causas dentro de Blockchain, así también conceptos de los riesgos que se encuentran en la misma tecnología.

CONCLUSIONES	Existe un riesgo latente para la criptografía asimétrica a un horizonte todavía lejano de posible rotura por computación cuántica, pero se ha considerado que es muy difícil de poner en contexto, porque sería realmente un cisne negro o gris. Sin embargo, sí que hay que considerar los riesgos que introducen las implementaciones deficientes que se pueden encontrar a día de hoy [21].
APORTE	Brinda información de amenazas y sus causas, también información bastante detallada de los riesgos que puede haber dentro de Blockchain.
AÑO DE PUBLICACIÓN	2020

Elaborado por: El investigador

Tabla 6 Ficha Bibliográfica 6

FICHA BIBLIOGRÁFICA	
TEMA	Comparación de plataformas para Smart Contracts basadas en Blockchain.
TESIS	El artículo realiza un estudio de las plataformas que soportan Smart Contracts haciendo un análisis y explicando sus características.
PROPÓSITO	Estudiar las principales características de Blockchain y Smart Contracts.

IDEAS CENTRALES	• Se realizó una comparación de plataformas para Smart Contracts basadas en Blockchain, utilizando una serie de criterios que permiten mostrar las características más relevantes de esta tecnología [22]. • Los objetivos principales de los Smart Contracts son: agregar mayor seguridad a los contratos tradicionales, reducir costos administrativos, reducir el tiempo asociado a este tipo de interacciones, evitar tener que usar un intermediario [22].
CONCEPTOS CLAVES	Contiene conceptos de Smart Contracts, Hash, Fork, Transacción, Redes P2P, Token.
CONCLUSIONES	Durante el transcurso del proyecto, se ha observado cómo esta tecnología está viva y su popularidad ha crecido enormemente, tanto es así que el ranking de las primeras veinte plataformas de CoinMarketCap ha cambiado radicalmente desde el día de comienzo de esta investigación [22]. Algo que sucede en este rubro es que las plataformas continúan reinventándose, buscando proporcionar lo que antes no hacían. Se observa cómo algunos cambios no son tan simples de impactar [22].
APORTE	Brinda información de diferentes Blockchain y explica que beneficios de Smart Contracts en cada una de ellas.
AÑO DE PUBLICACIÓN	2020

Elaborado por: El investigador

Tabla 7 Ficha Bibliográfica 7

FICHA BIBLIOGRÁFICA	
TEMA	Explorando la Blockchain de Ethereum y el desarrollo de Smart Contracts.
TESIS	Este proyecto pretende dar a conocer la tecnología Blockchain y también el uso y desarrollo de los Smart Contracts.
PROPÓSITO	Explicar sobre Blockchain y también sobre los Smart Contracts sus características y uso.
IDEAS CENTRALES	• Los Smart Contracts son justamente contratos como en el mundo real, pero la única diferencia es que estos son completamente digitales, de hecho, este contrato es un pequeño programa informático que es guardado dentro de una Blockchain [23]. • Los contratos en Solidity son similares a las clases en lenguajes orientados a objetos. Contienen datos persistentes en variables de estado y funciones que pueden modificar estas variables [23].
CONCEPTOS CLAVES	Contiene conceptos de Smart Contracts y propiedades que heredan.

CONCLUSIONES	Con esta prueba de concepto basada en privacidad médica se demuestra lo rápido que se puede integrar la tecnología hoy en día. Como el usuario final puede interactuar con los Smart Contracts y aprovechar todas las características que ofrecen como si una aplicación tradicional se tratará, pero asegurándose que sus datos están descentralizados y no hay una entidad o empresa que almacene los datos y no se conozca que pueden hacer con ellos [23].
APORTE	Brinda información de las características y estructura de los Smart Contracts.
AÑO DE PUBLICACIÓN	2018

Elaborado por: El investigador

2.2.4 Procesamiento y análisis de datos

De acuerdo con la información recolectada de temas similares con relación a la tecnología Blockchain y Smart Contracts se establece que:

- Blockchain es una tecnología con bastante potencial y se puede decir que no esta libre de amenazas, vulnerabilidades y riesgos.
- Así como existen riesgos también existen medidas que se pueden usar para prevenirlos.
- Los errores de programación al momento de implementar un programa en Blockchain o en el desarrollo de los Smart Contracts tienen un papel muy importante en la seguridad de estos sobre todo porque pueden ser explotados sino están bien implementados.

- Lo que más se puede destacar de Blockchain y los Smart Contracts es la descentralización y las transacciones no están sujetas al control de una sola entidad centralizada.
- Los Smart Contracts permiten a las partes realizar acuerdos de manera automática y sin necesidad de intermediarios.
- Los Smart Contracts se basan en la tecnología Blockchain y son inmutables transparentes y seguros, pero así mismo no están exentos de amenazas.
- Se requiere un cuadro con los riesgos de la tecnología Blockchain y buscar sus posibles soluciones.
- Los Smart Contracts son programables, lo que permite la creación de lógica personalizada y la automatización de tareas complejas.

CAPÍTULO III.- RESULTADOS Y DISCUSIÓN

3.1 Análisis y discusión

3.1.1 Riesgos de la tecnología Blockchain y la gestión de estos.

La tecnología Blockchain ha ganado popularidad en los últimos años debido a su capacidad para proporcionar un registro inmutable y transparente de transacciones. Sin embargo, a pesar de sus beneficios, el uso de Blockchain también plantea riesgos significativos para la privacidad de la información.

Tabla 8 Riesgos de Blockchain

RIESGO	DETALLE	GESTIÓN DEL RIESGO
Race attack	Se genera un doble gasto debido a que el comerciante acepta el pago como bueno antes de que la transacción haya sido confirmada, lo que le causa perjuicios económicos [21].	• Aumentar el tiempo de espera entre transacciones. • Utilizar servicios de pago confiables. • Asegurar una conexión de red confiable. • Actualizar y mejorar el protocolo de consenso.
Finney attack	Con la colaboración de un minero que incluye la transacción fraudulenta en un bloque. Igual que en el caso anterior, afecta al comerciante [21].	• Utilizar cadenas de bloques con alta tasa de hash. • Diversificar la minería. • Establecer alertas y monitoreo. • Mantener el software actualizado.
Vector76 attack	Un minero genera dos nodos con dos transacciones idénticas, una con mayor valor, y otra con menos valor para forzar la aceptación de la de mayor valor. Requiere sacrificar un bloque sin minar [21].	• Usar sistemas que no acepten transacciones con una sola confirmación. • Definir conexiones entrantes desde equipos reconocidos. • Las conexiones de salida del nodo deben ser monitoreadas.

Alternative history attack	Un atacante envía una transacción y seguidamente genera un fork con la transacción de vuelta. Consume mucho hashrate (recursos computaciones en el cálculo de hashes de bloques) [21].	• Mantener actualizado el software. • Aumentar la potencia del hash.
Dificultad de programar	Como triggers las cláusulas del contrato o condiciones del cambio de estado. Lenguaje natural versus lenguaje de programación [21].	• Documentar y comentar el código. • Utilizar recursos de aprendizaje.
Imposibilidad de cambiar a posteriori	Un Smart Contract sin provocar un fork de toda la Blockchain [21].	• Pruebas exhaustivas. • Contratos actualizados o flexibles. • Seguimiento y monitoreo.
Concentración de poder	En los coordinadores de los pools de mineros, pudiendo llegar incluso a superar el 51% y a hacerse con el control total de la Blockchain [21].	• Implementar algoritmos de consenso robustos. • Descentralizar el desarrollo y la gobernanza. • Incentivar la diversidad de nodos.
Seguridad de las credenciales	Existen riesgos de abuso de las credenciales tanto en la parte cliente (por el almacenamiento de la clave privada sin protección o incluso en la nube en servicios gestionados por intermediario lo que a veces provoca por vulnerabilidades que dichas claves sean expuestas) [21].	• Autenticación sólida. • Almacenamiento seguro de credenciales. • Uso de carteras seguras. • Monitoreo y detección de actividad sospechosa.
Almacenamiento creciente	Sin posibilidad de borrado: Por la propia naturaleza de Blockchain, la información se replica en todos los nodos, de modo que algunas Blockchains ocupan TB y TB de información, que seguirá incrementándose [21].	• Escalabilidad (fragmentación de la Cadena de Bloques, compresión de los datos, almacenamiento en la nube). • Eficiencia de acceso a datos históricos. • Riesgo de bifurcaciones.
Obsolescencia de la criptográfica	Por el incremento de la potencia de computación o desarrollo de nuevos algoritmos que minen la seguridad proporcionada por la criptografía de curva elíptica y	• Mantenerse actualizado con los avances criptográficos. • Seguir las mejores prácticas.

	los algoritmos usados actualmente [21].	• Mantener una postura proactiva. • Evaluar la seguridad criptográfica de terceros.

Elaborado por: El investigador

3.1.2 Capacidad de los Contratos Inteligentes, funciones y características.

Los Contratos Inteligentes son programas informáticos que se ejecutan en una Blockchain y están diseñados para automatizar y hacer cumplir acuerdos digitales de manera transparente, confiable y segura. Estos contratos poseen la capacidad de almacenar y ejecutar reglas y condiciones predefinidas, permitiendo así la transferencia de activos digitales o el intercambio de información sin necesidad de intermediarios.

Características de los Contratos Inteligentes [24]

- Inmutabilidad: Una vez creadas las instrucciones en el Blockchain, no se pueden modificar, ni siquiera por una autoridad judicial.
- Auto ejecución: una vez cumplidas las condiciones, la maquina sin intervención de un tercero automáticamente ejecuta las prestaciones acordadas.
- Rapidez en la ejecución del contrato.
- No requieren la intervención del Estado para su perfeccionamiento o regulación.
- Son seguros en la medida que la identificación de las partes sea estricta.

Ventajas de los Contratos Inteligentes [25]

- Autonomía: En los Smart Contracts, no es necesario que intervenga un tercero, como un abogado o un notario, para que den fe o seguridad a las partes.
- Ahorro de costes: La falta de intervención de personas en la ejecución del contrato reduce los costes.

- Seguridad: El uso de la tecnología Blockchain brinda seguridad y evita fraudes, ya que aporta transparencia y permite la trazabilidad de cada operación.
- Ahorro de tiempo: Los contratos inteligentes utilizan la automatización de procesos, por lo que disminuye el tiempo que se dedica a cada tarea, y las empresas tienen la oportunidad de poner el foco en desarrollar su negocio.
- Eliminación de errores: La automatización y el uso de la tecnología en este tipo de contratos supone que los errores se reduzcan en gran medida.
- Sostenibilidad: Los Smart Contracts contribuyen a un desarrollo sostenible. Con un contrato inteligente no son necesarios los desplazamientos, por tanto, se evita el consumo de combustible. Asimismo, no requiere el uso de oficinas o papel, por lo que se protege el medioambiente.

Aplicaciones de los Contratos Inteligentes [25]

- Confirmar la entrega de un producto: hace posible firmar un contrato por el que se confirma la entrega de un producto, de manera que, cuando el contrato inteligente tenga la certeza de que se ha producido dicha entrega, se debe pagar una determinada cantidad de dinero.
- Registrar títulos académicos: en recursos humanos, los Smart Contracts se pueden utilizar para registrar títulos académicos para que no exista ningún fraude en los currículums y se conozca con exactitud la formación de cada candidato, de manera que se adaptará mejor al puesto de trabajo ofertado.
- Firmar hipotecas o créditos: en el sector financiero, si se cumplen una serie de requisitos preestablecidos, por ejemplo, un ratio de solvencia del solicitante, el banco otorga el crédito de forma automática. De esta manera, los procesos de financiación son mucho más rápidos y baratos.
- Compensación económica: en el ámbito de los viajes, por ejemplo, un Smart Contract podría compensar al viajero automáticamente en caso de producirse

algún retraso en un vuelo o una cancelación. De esta forma, se evitan reclamaciones y trámites burocráticos en los que es necesario invertir tiempo y esfuerzo.

- Asimismo, en el sector de los seguros, si se produce un siniestro en un automóvil, la empresa aseguradora podría pagar automáticamente al asegurado, una vez cotejados todos los requisitos. En este caso, el ahorro de trámites podría suponer una reducción en las primas de los seguros.

- Registrar la propiedad o cambiarla: los contratos inmobiliarios se pueden convertir en contratos inteligentes que se ejecuten sin necesidad de intermediarios como abogados, notarios o consultores. Es el caso de los contratos de alquiler de viviendas, locales comerciales u otros inmuebles, así como el de la compraventa de activos inmobiliarios.

Problemas y desafíos de los Contratos Inteligentes

Al igual que cualquier otra tecnología, los contratos inteligentes no están exentos de problemas y desafíos. En realidad, el mercado está evolucionando y hay muchas empresas que buscan formas de adoptar la tecnología Blockchain. De hecho, la adopción de contratos inteligentes se ve obstaculizada por el hecho de que la tecnología Blockchain no está madura [26].

Los principales desafíos en el mundo de los Contratos Inteligentes incluyen los siguientes [26]:

- Curva de Adopción: La tecnología Blockchain, en la que se basan los Contratos Inteligentes, aún está en sus etapas iniciales. Esto afecta la adopción de los Contratos Inteligentes, ya que las empresas deben superar numerosos obstáculos para implementarlos, incluyendo la necesidad de establecer una red descentralizada con registros adecuados.

- Legalidades y Regulaciones: Cualquier implementación de Contratos Inteligentes debe cumplir con las regulaciones y leyes aplicables en el ecosistema y el lugar donde se ejecuten.
- No Estandarizado: Los Contratos Inteligentes aún no tienen estándares definidos, lo que ha llevado a múltiples enfoques y soluciones disponibles en línea.
- Curva de Aprendizaje: Tanto la tecnología Blockchain como los Contratos Inteligentes son complejos y requieren un alto grado de conocimiento técnico y legal para ser dominados. Los desarrolladores deben no solo ser capaces de codificar, sino también de comprender el aspecto legal del código que están escribiendo para el contrato. Además, los jueces y agencias legales también deben estar preparados para entender el código o buscar interpretaciones para tomar decisiones adecuadas.
- Complejidad del Ecosistema Empresarial: Los Contratos Inteligentes no son fáciles de integrar en sistemas empresariales complejos, ya que su implementación requiere una planificación cuidadosa y adaptación a las estructuras existentes.
- Privacidad de Datos: La inmutabilidad de Blockchain puede ser un problema en términos de cumplimiento de las leyes de privacidad de datos, como el GDPR. Es necesario abordar la privacidad sin comprometer los fundamentos de la tecnología Blockchain.

3.2 Desarrollo de la propuesta

Metodología

Para la gestión del proyecto se utilizó la metodología Kanban debido a su enfoque visual y flexible que permite una gestión eficiente del flujo de trabajo. Teniendo en cuenta el tiempo limitado que se tiene a disposición, se necesita una metodología que se pueda implementar rápidamente y que proporcione una visualización clara de las tareas pendientes y en progreso. Kanban permite priorizar y organizar las actividades de

manera efectiva, lo que ayudará a maximizar la productividad y asegurar un progreso constante hacia la finalización exitosa del proyecto de investigación.

Aplicación de la metodología

Para la aplicación de la metodología se utilizó el software Trello que tiene una interfaz intuitiva y fácil de usar, lo que permitirá empezar rápidamente a organizar las tareas en el tablero Kanban. Trello permite personalizar el tablero según las necesidades para representar las distintas etapas del proyecto. Además, Trello también permite colaborar en equipo en tiempo real, lo que facilita la comunicación y coordinación con los miembros del equipo.

Las tareas por realizar para el desarrollo de la propuesta son las siguientes:

1. Analizar las arquitecturas de Blockchain existentes.
2. Definir los requisitos funcionales y no funcionales para la arquitectura.
3. Investigar sobre los distintos lenguajes de programación compatibles con Contratos Inteligentes.
4. Selección de la arquitectura de Blockchain y lenguaje de programación compatibles con Contratos Inteligentes a utilizar.
5. Diseño de la arquitectura de Blockchain.
6. Implementación de Contratos Inteligentes.
7. Realizar pruebas de funcionalidad.

Flujo de trabajo

Usando el software Trello se generó el tablero Kanban con el diagrama de flujo de trabajo donde se establecieron la lista de tareas, las tareas en proceso y las tareas ya hechas del desarrollo del proyecto.

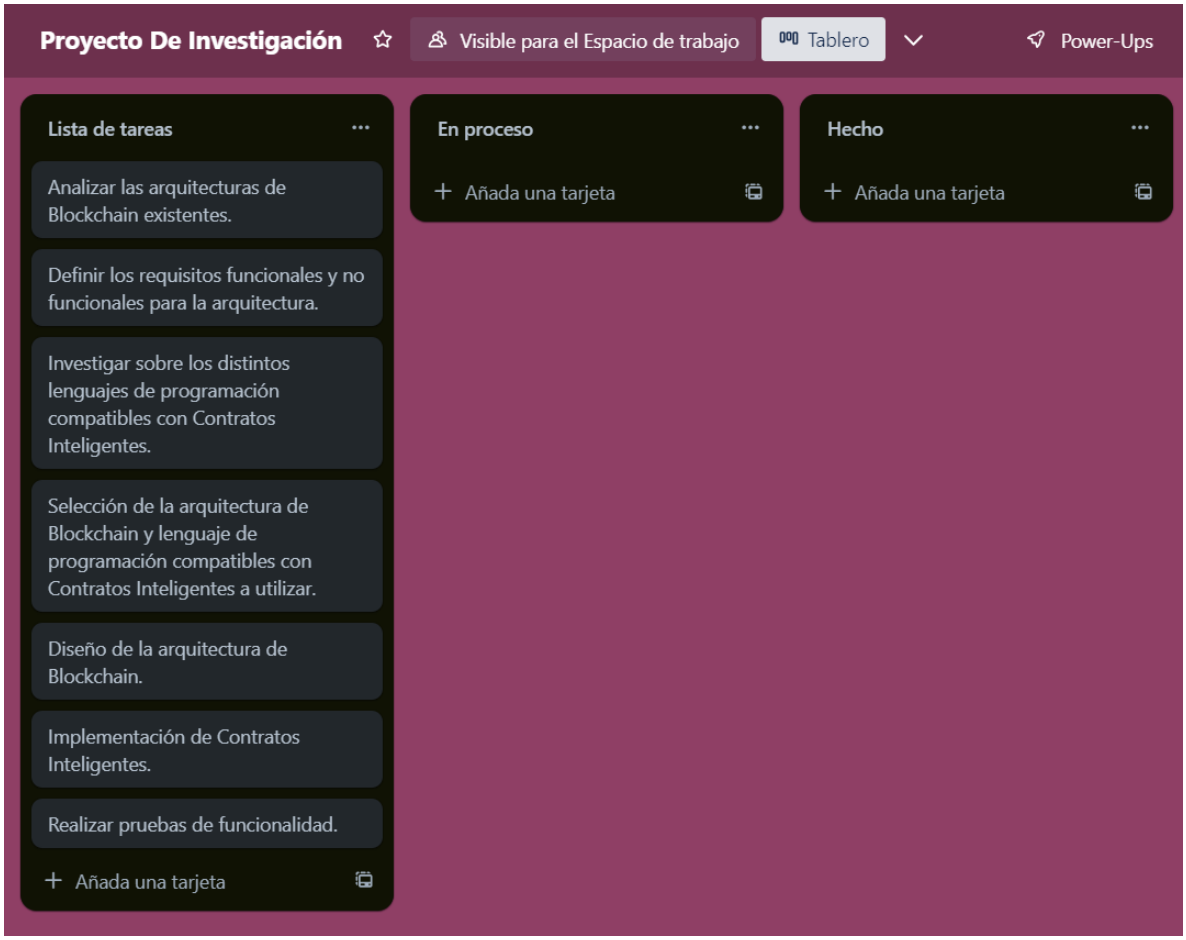


Figura 1 Tablero Kanban

Elaborado por: El investigador

La Figura 1 hace referencia al inicio de la fase de planificación del proyecto en la cual se puede observar la lista de tareas por realizar durante el desarrollo de la propuesta.

1. Analizar las arquitecturas de Blockchain existentes

Ethereum

Ethereum es una plataforma de cadena de bloques que permite a los usuarios desarrollar y ejecutar aplicaciones descentralizadas. Estas aplicaciones permiten a los usuarios realizar transacciones financieras sin tener que confiar en una entidad central. Esta plataforma de criptomoneda también ofrece contratos inteligentes y tokens ERC-20 que

se pueden usar para realizar transacciones seguras y confiables. Aunque Ethereum tiene muchos beneficios, también tiene algunas desventajas [27].

Ventajas de Ethereum [27]**:**

- Seguridad: La plataforma ha sido diseñada con un enfoque en la seguridad, lo que brinda a los usuarios tranquilidad en cuanto a la protección de sus fondos. Los contratos inteligentes y tokens ERC-20 también contribuyen a transacciones más seguras y confiables.
- Descentralización: Ethereum no depende de una entidad central, lo que permite a los usuarios realizar transacciones sin necesidad de confiar en intermediarios.
- Rapidez: Comparado con otras criptomonedas, Ethereum es conocido por ser más rápido en la realización de transacciones.
- Bajos costos: Las tarifas de transacción en Ethereum suelen ser inferiores a otras criptomonedas, lo que permite a los usuarios ahorrar dinero al realizar operaciones.
- Amplia funcionalidad: Ethereum ofrece una amplia gama de herramientas y funcionalidades que permiten desarrollar y ejecutar diversas aplicaciones descentralizadas.

Desventajas de Ethereum [27]**:**

- Alto costo de electricidad: Mantener la red de Ethereum requiere una considerable cantidad de electricidad, lo que puede resultar costoso para los usuarios.
- Falta de escalabilidad: La plataforma aún enfrenta desafíos para procesar una gran cantidad de transacciones simultáneamente, lo que limita su escalabilidad.
- Falta de privacidad: Ethereum es completamente transparente, lo que significa que las transacciones no son privadas y pueden ser vistas por cualquiera.
- Volatilidad: La criptomoneda de Ethereum puede ser extremadamente volátil, lo que representa un riesgo significativo para los usuarios que invierten en ella.

Hyperledger Fabric

Hyperledger Fabric es un Blockchain permisible o que requiere de permiso para poder participar. Este concepto está basado en control, donde se crean canales, para invitar a las empresas con las que se quiera llevar a cabo un trámite. HF no es un Blockchain público que esté disponible gratuitamente para todo el mundo, como Bitcoin o Ethereum. El concepto se basa en los primeros desarrollos de IBM y Digital Asset. El proyecto está dirigido principalmente a empresas que requieren Tecnología de Ledger Distribuido (TLD). El framework también ofrece la opción de utilizar contratos inteligentes (Smart Contracts), pero Hyperledger Fabric se le hace referencia a ellos como Chaincode [28].

Ventajas [28]

- Autorización: La membresía en un Blockchain permisible no es gratuita. Los usuarios deben registrarse y renunciar a su anonimato. Esto es importante para una empresa, porque usted quiere saber con quién está haciendo negocio, pero no quiere revelar los secretos de su empresa y los procesos internos de su tecnología a la competencia.

- Escalabilidad, personalización y confianza: La estructura es modular. Esto permite una mayor escalabilidad. Dado que no todos los operadores de nodos tienen que estar incluidos en una transacción, el nivel de confianza y verificación puede reducirse. Sólo las personas directamente implicadas en una operación en específico obtienen información.

- Necesidad de saber: La competencia es un tema importante en la gestión empresarial y corporativa. Las empresas pueden negociar precios diferentes. Lo ideal, sin embargo, es que no todo el mundo pueda experimentar esto. Con esta función de Hyperledger Fabric, las empresas pueden hacer que determinadas operaciones sean visibles para un grupo determinado.

- Modularidad y Plug-In: Como ya se ha mencionado, Hyperledger Fabric trabaja con SDKs y tiene una estructura modular. Esto permite a las empresas desarrollar una red de acuerdo a sus ideas y necesidades. Esto hace que el marco

sea tan versátil y adecuado para todas las industrias imaginables. Los componentes existentes también se pueden combinar fácilmente con otras soluciones de Blockchain. Por lo tanto, no todo tiene que ser reconstruido.

- Seguridad: Hyperledger Fabric utiliza módulos de seguridad de hardware (MSH) para proteger y gestionar claves e identidades digitales. MSH mejora la gestión de identidades y protege los datos confidenciales.

Cardano

Cardano fue creado como una alternativa a Ethereum y es el primer protocolo de Blockchain descentralizado revisado por pares que utiliza un enfoque científico. Los desarrolladores de Cardano pretenden crear una plataforma de Blockchain que pueda procesar más transacciones a un bajo costo. Al mismo tiempo, buscan proteger los datos de los usuarios integrando la tecnología de libro mayor distribuido y la infraestructura de contratos inteligentes [29].

La Blockchain de Cardano permite que se celebren contratos inteligentes, se creen aplicaciones y protocolos descentralizados y se envíen y reciban fondos de forma instantánea con comisiones mínimas. El token de utilidad ADA de Cardano se utiliza como una transferencia de valor como muchos otros tokens. Pero se diferencia de otras criptomonedas por sus funcionalidades. Los comerciantes del grupo de staking lo utilizan en el sistema de staking (bloqueo de monedas para obtener recompensas) para mantener la seguridad del protocolo. Aquellos que hacen staking con sus tokens ADA en la Blockchain los utilizan para verificar las transacciones [29].

Ventajas y Desventajas

Entre las ventajas de Cardano se encuentran un gran equipo de desarrollo, un potencial de escalabilidad ilimitado, transacciones rápidas y económicas con la criptomoneda ADA y un mecanismo de consenso justo, además de la posibilidad de crear aplicaciones descentralizadas [29].

Uno de los inconvenientes de Cardano es el hecho de que la Blockchain está todavía en desarrollo. Necesita mejorar sus problemas de escalabilidad, ya que solo es capaz de procesar 257 transacciones por segundo (TPS). Por otro lado, tiene problemas con la sincronización de la billetera oficial y la conectividad de la red. Otro problema es el ataque de doble gasto o del 51 %, ya que todavía existe el peligro de que los endosantes de entrada aprueben el mismo conjunto de transacciones de dos líderes de bloques diferentes [29].

EOS

EOS.IO es una plataforma descentralizada, que hace uso de tecnología Blockchain, cuyos principales atractivos residen en la capacidad que tiene tanto para desarrollar y ejecutar aplicaciones descentralizadas (Dapps), así como por la capacidad para procesar un gran número de operaciones, dentro de su plataforma, y a muy alta velocidad [30].

Ello también ha hecho que su token principal, EOS, gracias a la arquitectura de la plataforma, sea capaz de alcanzar una gran escalabilidad, lo que ha convertido a EOS.IO en rival directo de Ethereum y Cardano [30].

Ventajas [30]**:**

- Desarrollo sencillo de Dapps: Permite a los desarrolladores crear aplicaciones descentralizadas de manera más fácil con una amplia gama de posibilidades.
- Alta velocidad de procesamiento: Capacidad para manejar un gran número de operaciones a una velocidad comparable a gigantes como Visa o PayPal.
- Equipo experimentado: Cuenta con un equipo reconocido y experimentado en el proyecto.
- Adaptabilidad: La plataforma es fácilmente adaptable y modificable para actualizarla ante posibles ataques o fallos en el sistema.
- Descentralización: Utiliza el protocolo DPOS, que garantiza equidad y bajas o nulas comisiones en las operaciones.

Desventajas [30]**:**

- Competencia con rivales fuertes: Enfrenta la competencia directa de Ethereum y Cardano, dos plataformas con grandes expectativas y amplio desarrollo comunitario.
- Dependencia del Smart Contract ERC-20 de Ethereum: Aunque ha logrado independencia en gran medida, todavía depende del Smart Contract ERC-20 de Ethereum en algunos aspectos.

Cada arquitectura presenta sus propias ventajas y desafíos, lo que destaca la relevancia de una evaluación precisa y a medida para cada proyecto. La flexibilidad, escalabilidad, seguridad y adaptabilidad emergen como factores fundamentales en el proceso de toma de decisiones. Al considerar las arquitecturas desde esta perspectiva, se establece un sólido fundamento para seleccionar una estructura que se alinee con los objetivos y requisitos específicos de cada caso.

2. Definir los requisitos funcionales y no funcionales para la arquitectura.

Requisitos funcionales:

- Registro y validación de transacciones: La arquitectura debe permitir el registro seguro y verificable de transacciones en la Blockchain utilizando Contratos Inteligentes.
- Ejecución automática de contratos: Los Contratos Inteligentes deben poder ejecutarse automáticamente y garantizar que las condiciones y términos acordados se cumplan de manera confiable.
- Privacidad y confidencialidad: Los requisitos de privacidad deben ser considerados, como la capacidad de realizar transacciones de forma confidencial o proteger ciertos datos del acceso no autorizado.

Requisitos no funcionales:

- Seguridad: La arquitectura debe garantizar la integridad, la autenticidad y la inmutabilidad de los datos almacenados en la Blockchain, así como la resistencia a ataques maliciosos.
- Eficiencia y rendimiento: La arquitectura debe ser eficiente en términos de consumo de recursos, como capacidad de procesamiento y uso de almacenamiento, para garantizar una respuesta rápida y un rendimiento óptimo.
- Usabilidad: La arquitectura debe ser fácil de usar y comprensible, permitiendo una interacción intuitiva con los contratos inteligentes y la Blockchain.

Figura 2 Tablero Kanban Actualizado

Elaborado por: El investigador

Una vez realizadas las primeras tareas, se puede observar en la Figura 2 que las dos primeras tareas pasaron a la columna de “Hecho” así mismo la tercera tarea pasó a la

columna de “En proceso”, esto se debe a la metodología Kanban que permite tener un registro sincronizado con las tareas que se van cumpliendo y las que están por hacer.

3. Investigar sobre los distintos lenguajes de programación compatibles con Contratos Inteligentes.

En la búsqueda de un lenguaje de programación que permita el desarrollo de Contratos Inteligentes, surge la necesidad de investigar sobre los distintos lenguajes de programación compatibles con esta tecnología de Blockchain. Cada lenguaje posee sus propias características, ventajas y desafíos que influyen directamente en la eficiencia y el alcance de los Contratos Inteligentes que se pueden desarrollar.

Solidity

Solidity es un lenguaje de programación de alto nivel, con enfoque en orientación a objetos, especialmente creado para desarrollar Smart Contracts en la plataforma Ethereum. Se ha convertido en uno de los lenguajes más ampliamente utilizados para la creación de contratos inteligentes en Ethereum debido a su sintaxis sencilla y su habilidad para ser compilado en bytecode, lo que permite su ejecución en la red Ethereum. Su diseño se ha inspirado en otros lenguajes de programación populares como C++, Python y JavaScript [31].

Con Solidity, los desarrolladores tienen la capacidad de crear Smart Contracts que pueden gestionar el intercambio de criptomonedas, la propiedad de activos digitales y otras transacciones similares. Estos contratos inteligentes, escritos en Solidity, poseen una característica importante: son inmutables, lo que significa que una vez creados, no pueden ser modificados. Además, se ejecutan automáticamente, lo que los convierte en una excelente opción para aplicaciones descentralizadas que requieren niveles altos de seguridad y transparencia [31].

Utilidad de Solidity

Algunos ejemplos de aplicaciones que pueden ser creadas utilizando Solidity y contratos inteligentes en Ethereum incluyen [31]:

- Plataformas de intercambio de criptomonedas descentralizadas.
- Sistemas de votación y toma de decisiones descentralizados.
- Plataformas de préstamos y créditos descentralizados.
- Sistemas de seguimiento de la cadena de suministro descentralizados.
- Sistemas de gestión de identidades descentralizados.

Funcionalidad de Solidity

Los contratos inteligentes en Ethereum son ejecutados por una máquina virtual que interpreta el código escrito en Solidity. Esta máquina virtual se ejecuta en los nodos de la red Ethereum y es la responsable de verificar y ejecutar los Smart Contracts [31].

Los contratos inteligentes desarrollados en Solidity son concebidos con un enfoque en la seguridad y la transparencia. Una vez que son desplegados en la red Ethereum, se vuelven inmutables, lo que significa que no pueden ser alterados. Esta característica asegura que los usuarios de la red puedan confiar en que los contratos se ejecutarán de acuerdo a cómo fueron programados [31].

Solidity se basa en la sintaxis de otros lenguajes de programación como C++, Python y JavaScript. Además, incluye características de programación orientada a objetos, como herencia, encapsulamiento y polimorfismo, lo que capacita a los desarrolladores para crear contratos inteligentes complejos [31].

Vyper

Vyper es un lenguaje de programación basado en Python que se enfoca en la creación de Smart Contracts para la máquina virtual de Ethereum (EVM). Al estar fundamentado en Python, se destaca por su facilidad de desarrollo de aplicaciones descentralizadas (dApps) para aquellos familiarizados con este lenguaje. Aprovecha las robustas herramientas de depuración disponibles para Python [32].

Entre los principales objetivos de Vyper se encuentran [32]:

- Ofrecer una mejor seguridad, ya que debería ser posible y natural construir contratos inteligentes seguros en Vyper.
- Simplicidad del lenguaje y del compilador: El lenguaje y la implementación del compilador deben esforzarse por ser simples.
- Mejorar la auditoría del código. El código de Vyper debe ser lo más legible posible. Además, debe ser lo más difícil posible escribir código engañoso. La simplicidad para el lector es más importante que la simplicidad para el escritor, y la simplicidad para los lectores con poca experiencia previa con Vyper (y poca experiencia previa con la programación en general) es particularmente importante.
- Permite una mejor comprobación de límites y desbordamiento, especialmente en los accesos a arrays y en la aritmética.
- Soporte para enteros con signo y números decimales de punto fijo.
- Decidibilidad: Es posible calcular un límite superior preciso para el consumo de gas de cualquier llamada a una función de Vyper.
- Tipificación fuerte, para evitar los problemas de seguridad que el tipado dinámico puede traer.
- Código del compilador pequeño y comprensible.
- Soporte limitado para funciones puras: Cualquier cosa marcada como constante no puede cambiar el estado, lo que ayuda a mejorar la seguridad.

Rust

Rust es un lenguaje de programación compilado, versátil y multiparadigma que tuvo sus inicios como parte del proyecto Mozilla y actualmente es gestionado por la Rust Foundation. Su principal enfoque radica en proporcionar un alto nivel de seguridad, al punto de ser considerado uno de los lenguajes de programación más seguros para el desarrollo de aplicaciones [32].

El desarrollo de Rust es completamente abierto, lo que significa que se fomenta la participación y contribución de la comunidad. Gracias a este enfoque, Rust ha sido ampliamente adoptado en una variedad de proyectos, abarcando desde el navegador web Mozilla Firefox, el kernel de GNU/Linux y, por supuesto, en el ámbito de las criptomonedas.

En el mundo de las criptomonedas, Rust ha encontrado aplicaciones significativas. Por ejemplo, es utilizado para construir nodos de referencia para Bitcoin, como Electrum-RS, y para expandir las capacidades de Smart Contracts en Bitcoin a través de miniscript. Además, Rust es empleado en otras blockchains como Near, donde es el lenguaje por defecto, y también en Ethereum, donde se utiliza para desarrollar un SDK para la EVM (Ethereum Virtual Machine). Además, Rust es utilizado en Substrate, un marco de desarrollo para entornos Polkadot/Kusama [32].

Ride

Ride es un lenguaje de programación funcional que ha sido diseñado para facilitar el desarrollo de contratos inteligentes y aplicaciones descentralizadas (dApps) en la plataforma de Waves Blockchain. En Waves, los contratos inteligentes son escritos en Ride y pueden ser asignados tanto a cuentas como a tokens (activos). Por ejemplo, al asignar un script a una cuenta, esta se convierte en una dApp o cuenta inteligente, mientras que al asignar un script a un activo, este se transforma en un activo inteligente. La simplicidad y facilidad de uso de Ride lo hacen una opción conveniente para desarrollar soluciones inteligentes en la Blockchain de Waves [32].

Así, la funcionalidad de los scripts depende de su tipo [32]:

- El script de la dApp permite definir funciones invocables que pueden realizar diversas acciones en la Blockchain y una función de verificación que permite o deniega las transacciones y órdenes que se envían en nombre de la cuenta de la dApp.

- El script de la cuenta permite o deniega las transacciones y órdenes que se envían en nombre de la cuenta.
- El script de los activos permite o deniega las transacciones relacionadas con el activo.

C++

El lenguaje de programación C++ también se utiliza como lenguaje para desarrollar Smart Contracts en EOS. Los desarrolladores tienen la posibilidad de crear contratos inteligentes en EOSIO utilizando C++ como su lenguaje de programación. Para facilitar este proceso, el kit de herramientas de desarrollo de contratos EOSIO, conocido como EOSIO.CDT, proporciona las bibliotecas y herramientas necesarias para construir un Contrato Inteligente. [32].

Para desplegar el Contrato Inteligente en la Cadena de Bloques de EOS, se utiliza la herramienta eosio-cpp para compilar el código del contrato. Durante el proceso de compilación, se generan dos archivos: uno en formato Web Assembly (WASM) y otro archivo correspondiente de Interfaz Binaria de Aplicación (ABI) [32].

4. Selección de la arquitectura de Blockchain y lenguaje de programación compatibles con Contratos Inteligentes a utilizar.

Se utilizará el Blockchain de Ethereum y el lenguaje de programación Solidity para el diseño y desarrollo de la arquitectura debido a varias razones.

Tabla 9 Selección de Blockchain y Lenguaje de Programación

SELECCIÓN DE BLOCKCHAIN Y LENGUAJE DE PROGRAMACIÓN	
Blockchain:	Ethereum
Razones:	• Ethereum es una plataforma Blockchain consolidada y ampliamente adoptada en la actualidad. • Ofrece estabilidad y confiabilidad para el

	proyecto. • Proporciona características avanzadas como la ejecución eficiente de Contratos Inteligentes y escalabilidad.
Lenguaje de Programación:	Solidity
Razones:	• Solidity es el lenguaje de programación nativo de Ethereum. • Diseñado específicamente para escribir Contratos Inteligentes. • Aprovecha su sintaxis y características especializadas para la eficiente y segura programación de Contratos Inteligentes.
Ventajas adicionales:	• Ethereum y Solidity cuentan con una comunidad activa de desarrolladores. • Amplia gama de recursos y soporte disponibles. • Facilitan el aprendizaje y la colaboración de cualquier proyecto.

Elaborado por: El investigador

5. Diseño de la arquitectura de Blockchain.

Para el diseño de la arquitectura de Blockchain utilizando Contratos Inteligentes se empleó un sistema de votación basado en Ethereum y Solidity que representa una solución innovadora y segura para garantizar la transparencia, integridad y confiabilidad en los procesos electorales y de los datos. Al aprovechar la tecnología Blockchain, se puede crear un entorno descentralizado y resistente a la manipulación, donde cada voto es registrado de manera inmutable en la cadena de bloques. Utilizando el lenguaje de programación Solidity, se podrá desarrollar contratos inteligentes que automatizan y ejecutan las reglas y lógica del sistema de votación, asegurando la correcta contabilización de los votos y garantizando la confidencialidad y anonimato de los votantes. Este enfoque tiene el potencial de superar las limitaciones y desafíos

asociados con los sistemas de votación tradicionales, brindando transparencia, seguridad y confianza en los procesos electorales.

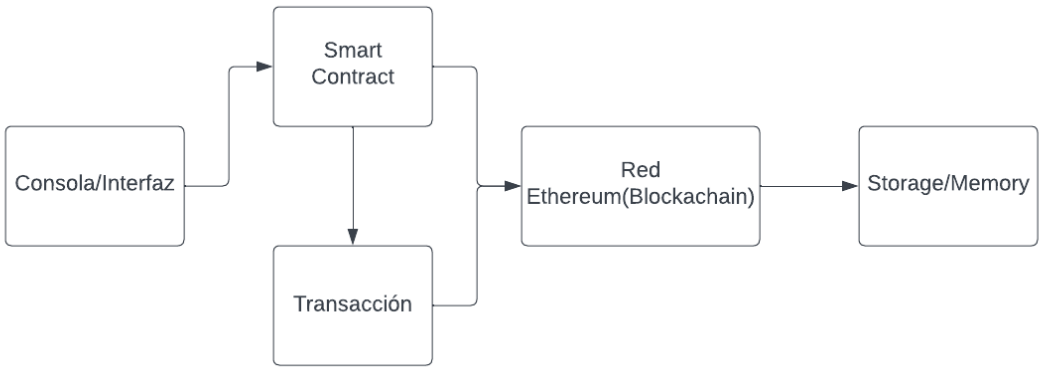


Figura 3 Diseño de la arquitectura

Elaborado por: El investigador

En la figura 3 se puede observar el diseño de la arquitectura la cual consta de los siguientes componentes:

1. Consola: este componente es el responsable de interactuar con las funciones del Contrato Inteligente, en la cual se registrarán las acciones principales del sistema.
2. Smart Contract: los Contratos Inteligentes se usarán para gestionar diferentes aspectos del sistema de votación, como la creación de candidatos, la emisión de votos y el recuento de votos
3. Transacción: este es el proceso fundamental en el funcionamiento del Contrato Inteligente en la Blockchain, esta transacción es validada y firmada por la Blockchain lo que garantiza la autenticidad e integridad.
4. Blockchain (Ethereum): la red Ethereum es la infraestructura que permite la ejecución de los Contratos Inteligentes, también proporciona la capacidad de

mantener un registro inmutable de transacciones en la cadena de bloques y asegura la integridad de los datos.

5. Storage/Memory: se utiliza un sistema de almacenamiento para mantener los datos relevantes para el sistema de votación, como la información de los candidatos, los resultados de las votaciones y los votos emitidos. Puede ser una base de datos o un sistema de almacenamiento distribuido que asegure la integridad y confidencialidad de los datos.

Herramientas a utilizar:

- Google Chrome
- Remix
- Metamask
- Etherscan
- Github

Google Chrome

Para el proyecto es necesario descargar el navegador Google Chrome debido a que el software Remix que es en donde se programarán los contratos inteligentes funciona solo en este navegador.

Remix

Remix es una herramienta en línea y gratuita desarrollada por Ethereum que se utiliza para escribir, probar y desplegar contratos inteligentes en la red Ethereum. Es una interfaz de desarrollo integrada (IDE) que proporciona un entorno amigable para desarrollar contratos inteligentes en la plataforma Ethereum.

Características principales de Remix:

- Editor de código.

- Compilador de Contratos Inteligentes.
- Permite depurar y probar contratos en un entorno simulado.
- Se puede interactuar directamente con contratos que hayan sido desplegados en la red de Ethereum.
- Una vez probado y depurado el Contrato Inteligente, se lo puede desplegar directamente en la red de Ethereum.

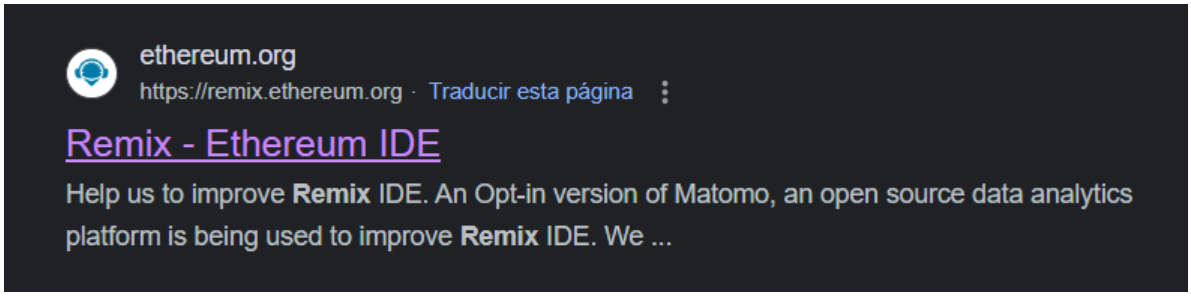


Figura 4 URL de la herramienta Remix

Elaborado por: El investigador

Una vez dentro del sitio se puede observar la interfaz de desarrollo mostrada a continuación.

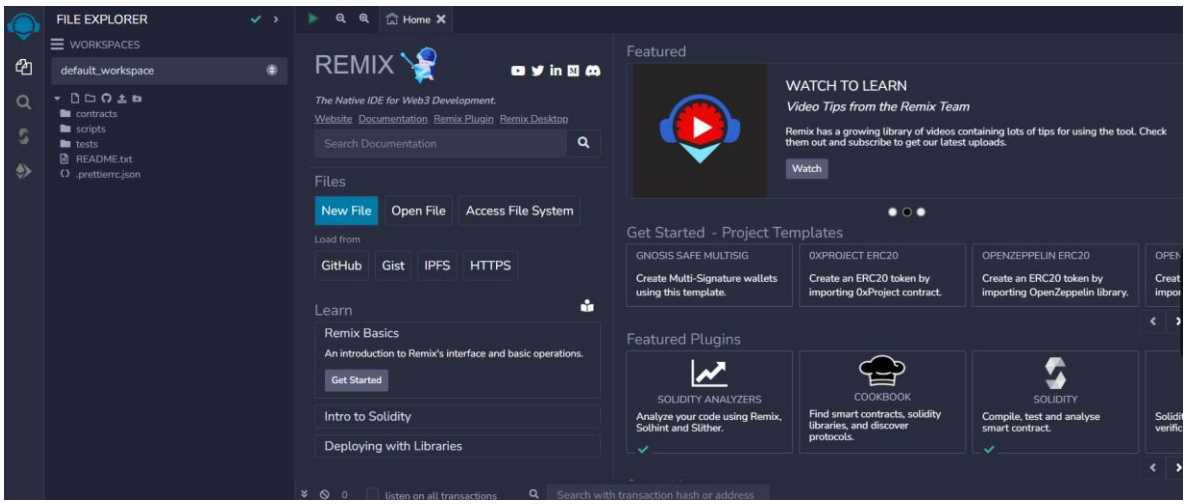


Figura 5 Remix IDE

Elaborado por: El investigador

Metamask

Metamask es una billetera digital en la que sus usuarios pueden almacenar y gestionar criptomonedas, tokens de diferentes Blockchains. Metamask es una extensión de distintos navegadores.

Metamask es una herramienta esencial para el proyecto puesto que permite interactuar con los Contratos Inteligentes a través de transacciones con el ecosistema de Ethereum.

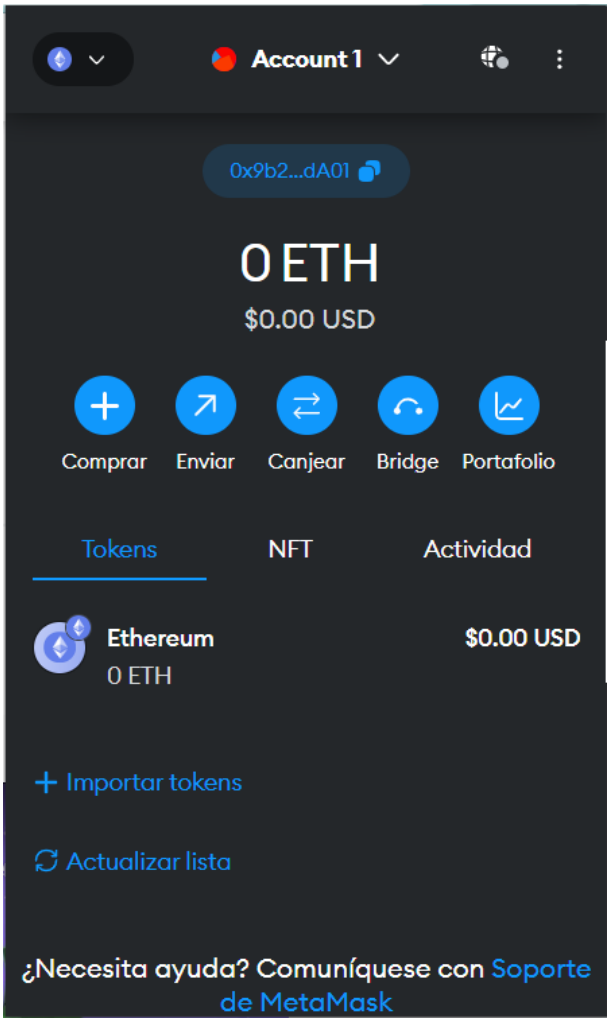


Figura 6 Billetera Digital Metamask

Elaborado por: El investigador

Etherscan

Esta herramienta sirve para proporcionar a los usuarios una forma fácil de explorar, rastrear y verificar las transacciones, contratos y otros datos relacionados con la Blockchain de Ethereum.

Github

Puesto a que la herramienta Remix es un IDE que funciona de manera local es necesario el uso de Github como una forma para almacenar y compartir los Contratos Inteligentes desarrollados en esta plataforma.

6. Implementación de los Contratos Inteligentes.

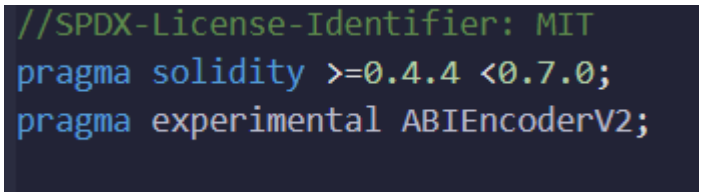


Figura 7 Librerías

Elaborado por: El investigador

Librerías a utilizar para programar en Solidity.

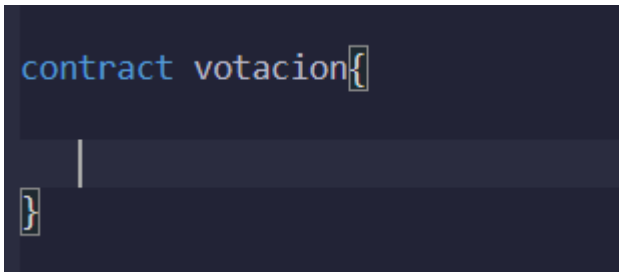


Figura 8 Creación del contrato

Elaborado por: El investigador

Para empezar a desarrollar el Contrato que contiene el sistema de votación, en el IDE de Remix se crea el contrato como se muestra en la Figura 8.

Este Contrato Inteligente va a tener una serie de reglas y lógica programada que van a ayudar a gestionar la votación. Cuando vez que se registre un voto el sistema interactuará con el Contrato Inteligente.

```
//Direccion del propietario del contrato
address owner;

//constructor
constructor () public{    887369 gas 865600 gas
    owner = msg.sender;
}
```

Figura 9 Asignación de propietario del contrato

Elaborado por: El investigador

La Figura 9 se muestra que en el constructor mediante la función msg.sender se asigna la dirección de la persona que despliega el contrato a la variable owner que es de tipo address.

```
//Relacion entre el nombre del candidato y el hash de sus datos personales
mapping (string=>bytes32) ID_Candidato;

//Relacion entre el nombre del candidato y el numero de votos
mapping (string=>uint) votos_candidato;
```

Figura 10 Relaciones de datos

Elaborado por: El investigador

Para poder ejecutar las distintas funciones del sistema se necesita hacer un mapping como se muestra en la Figura 10 relacionando el nombre del candidato con sus datos personales, así mismo de hace un mapping del nombre del candidato con el número de votos esto sirve para el conteo de votos.

```
//Lista para almacenar los nombres de los candidatos
string [] candidatos;

//Lista de los hashes de la identidad de los votantes
bytes32 [] votantes;
```

Figura 11 Listas dinámicas

Elaborado por: El investigador

En el código que se muestra en la Figura 11 la primera lista va a servir para almacenar a los candidatos que se registraron en las votaciones, la lista de los votantes es de tipo byte32 debido a que el voto es secreto por lo cual la única información que se va a almacenar seria la dirección de la billetera digital de la cual vota la persona y esta dirección pasaría a ser la identidad del votante.

```
//Funcion para presentarse a las elecciones
function Representar(string memory _nombrePersona, uint _edadPersona, string memory _idPersona) public{

    //Hash de los datos del candidato
    bytes32 hash_Candidato = keccak256(abi.encodePacked(_nombrePersona, _edadPersona, _idPersona));

    //Almacena el hash de los datos del candidato ligados a su nombre
    ID_Candidato[_nombrePersona] = hash_Candidato;

    //Almacena el nombre del candidato
    candidatos.push(_nombrePersona);

}
```

Figura 12 Función para presentarse a la votación

Elaborado por: El investigador

Para esta función Representar se necesita pasar 3 parámetros que son el nombre de la persona de tipo String, la edad de la persona de tipo entero y el id de la persona de tipo String.

El primer paso mediante la función keccak256 que sirve para calcular el hash de los datos del candidato. Después ese mismo hash se debe almacenar a los datos del candidato que estén ligados a su nombre.

Con la función push se almacena el nombre del candidato dentro del array.

```
//Permite visualizar las personas que se han presentado como candidatos a las votaciones
function VerCandidatos() public view returns(string[] memory){    infinite gas
    //Devuelve la lista de los candidatos presentados
    return candidatos;
}
```

Figura 13 Función para ver los candidatos

Elaborado por: El investigador

En la Figura 13 se puede observar el código de la función que permite ver la lista de personas que se han presentado como candidatos a las votaciones.

```
//Funcion para poder votar
function Votar(string memory _candidato) public{    infinite gas

    //Hash de la direccion de la persona que ejecuta esta funcion
    bytes32 hash_Votante = keccak256(abi.encodePacked(msg.sender));
    //Verificamos si el votante ya ha votado
    for(uint i=0; i<votantes.length; i++){
        require(votantes[i]!=hash_Votante, "Ya votaste");
    }
    //Almacena el hash del votante dentro del array de votantes
    votantes.push(hash_Votante);
    //Añade un voto al candidato seleccionado
    votos_candidato[_candidato]++;
}
```

Figura 14 Función para votar

Elaborado por: El investigador

El código de la función mostrada en la Figura 14 sirve para votar por un candidato, el hash del votante es la dirección con la que el usuario se conecta para votar, el ciclo recorre la lista de los votantes para evitar que la persona que yo votó pueda votar de nuevo.

```
//Dado el nombre de un candidato devuelve el numero de votos que tiene
function VerVotos(string memory _candidato) public view returns(uint){
    //Devolviendo el numero de votos del candidato _candidato
    return votos_candidato[_candidato];
}
```

Figura 15 Función para ver los votos

Elaborado por: El investigador

Esta función va a permitir ver los votos que tiene un candidato por medio de su nombre.

```
//Funcion auxiliar que transforma un uint a un string
function uint2str(uint _i) internal pure returns (string memory _uintAsString) {
    if (_i == 0) {
        return "0";
    }
    uint j = _i;
    uint len;
    while (j != 0) {
        len++;
        j /= 10;
    }
    bytes memory bstr = new bytes(len);
    uint k = len - 1;
    while (_i != 0) {
        bstr[k--] = byte(uint8(48 + _i % 10));
        _i /= 10;
    }
    return string(bstr);
}
```

Figura 16 Función auxiliar

Elaborado por: El Investigador

El código mostrado en la Figura 16 es una función auxiliar para transformar un uint a String que se va a ocupar en la siguiente función.

```
//Ver los votos de cada uno de los candidatos
function VerResultados() public view returns(string memory){    infinite gas
    //Guarda en una variable string los candidatos con sus respectivos votos
    string memory resultados="";

    //Recorre el array de candidatos para actualizar el string resultados
    for(uint i=0; i<candidatos.length; i++){
        //Actualiza el string resultados y añadimos el candidato que ocupa la posicion "i" del array candidatos
        //y su numero de votos
        resultados = string(abi.encodePacked(resultados, "(", candidatos[i], ", ", uint2str(VerVotos(candidatos[i])), ") -----"));
    }

    //Devolver los resultados
    return resultados;
}
```

Figura 17 Función para ver los votos de los candidatos

Elaborado por: El investigador

En esta Figura 17 se observa el código que sirve para ver los votos que van teniendo los candidatos durante las votaciones, esta función recorre el array de candidatos y actualiza la cantidad de votos a su favor para devolver la cantidad de votos de cada uno.

```
//Proporcionar el nombre del candidato ganador
function Ganador() public view returns(string memory){    infinite gas

    //La variable ganador contendra el nombre del candidato ganador
    string memory ganador= candidatos[0];
    bool flag;

    //Recorre el array de candidatos para determinar el candidato con un numero de votos mayor
    for(uint i=1; i<candidatos.length; i++){

        if(votos_candidato[ganador] < votos_candidato[candidatos[i]]){
            ganador = candidatos[i];
            flag=false;
        }else{
            if(votos_candidato[ganador] == votos_candidato[candidatos[i]]){
                flag=true;
            }
        }
    }

    if(flag==true){
        ganador = "¡Hay empate entre los candidatos!";

    }
    return ganador;
}
```

Figura 18 Función para ver el ganador de las votaciones

Elaborado por: El investigador

Esta función sirve para ver el String del candidato con mas votos de igual manera si después de recorrer el array y llegase a haber un empate en votos esta función avisará sobre esto.

7. Realizar pruebas de funcionamiento

Estas pruebas se dividirán en dos fases: la primera, se llevó a cabo pruebas a nivel local utilizando la red de Remix, esta etapa permitirá una evaluación temprana de la funcionalidad del sistema en un entorno controlado. Posteriormente, se ejecutará una segunda fase de pruebas, utilizando una red de Ethereum de prueba. Esta fase se encargará de someter el sistema a condiciones más cercanas a la realidad, explorando su rendimiento y confiabilidad.

Pruebas de funcionamiento de manera local

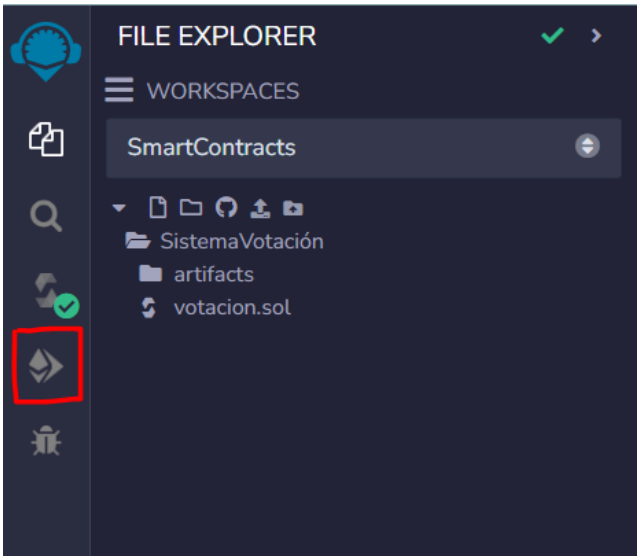


Figura 19 Remix Deploy

Elaborado por: El investigador

Para poder desplegar el Contrato Inteligente en la Figura 19 se muestra la pestaña de Deploy.

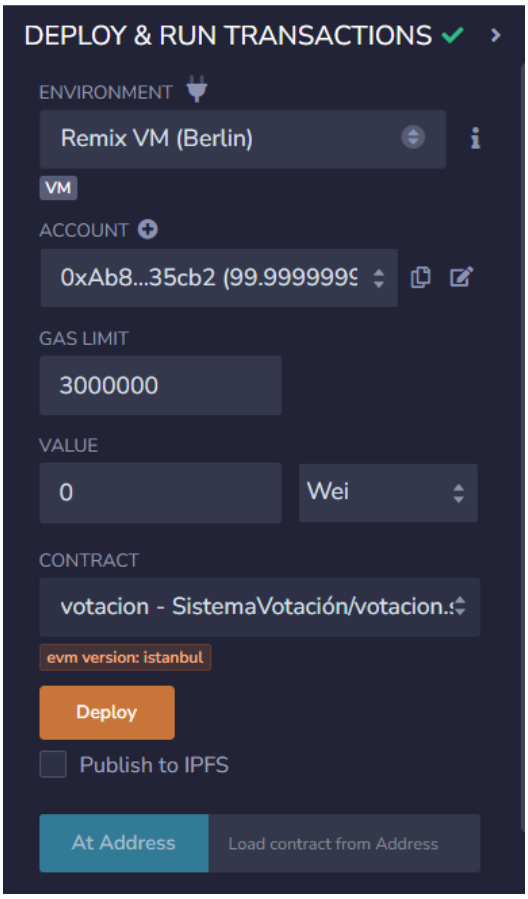


Figura 20 Consola para ejecutar las funciones

Elaborado por: El investigador

En la Figura 20 se muestra la consola para hacer el despliegue del Contrato Inteligente "votacion" así mismo se puede observar en el campo "Environment" que es el entorno de trabajo en este caso como se va a realizar la prueba de manera local se utiliza una maquina virtual llamada Remix VM (Berlín) la cual viene seleccionada por defecto.

En el campo "Account" se escoge la cuenta o dirección de prueba en la cual se va a desplegar el contrato y la que servirá de intermediario para realizar las transacciones.

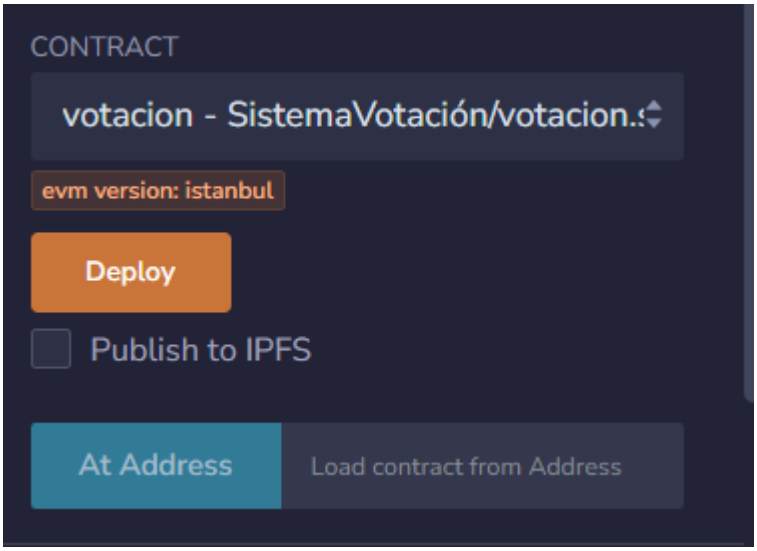


Figura 21 Desplegar el Contrato

Elaborado por: El investigador

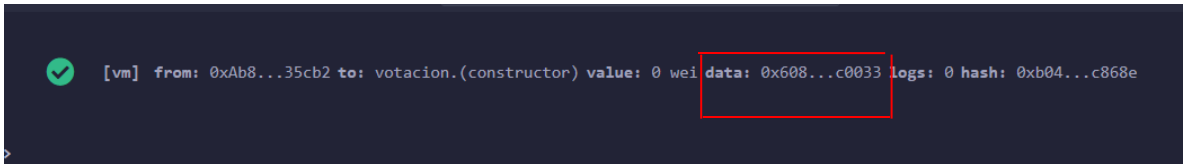


Figura 22 Despliegue exitoso del contrato

Elaborado por: El investigador

Una vez desplegado el Contrato Figura 22 se puede ver en la Figura 23 como el despliegue de este fue exitoso. Esto genera la firma del Contrato Inteligente en la red de Ethereum con la dirección del propietario es decir la persona que desplegó el Contrato.

Figura 23 Consola para interactuar con las funciones del Contrato

Elaborado por: El investigador

Una vez desplegado el Contrato Inteligente se podrá interactuar con las funciones del sistema que son las reglas establecidas del Contrato, mediante la consola que se muestra en la Figura 23.

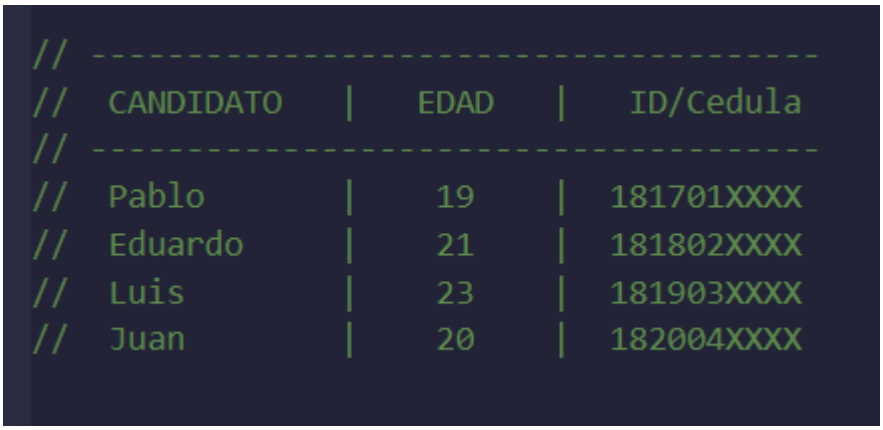


Figura 24 Candidatos

Elaborado por: El investigador

En la Figura 24 se puede observar los candidatos de prueba que se van a registrar para las votaciones.

Figura 25 Cargar los candidatos

Elaborado por: El investigador

Para cargar correctamente a un candidato hay que poner los parámetros que se programaron en la función.

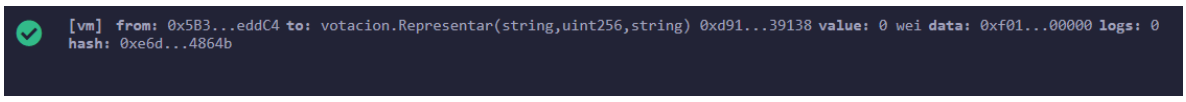


Figura 26 Transacción exitosa

Elaborado por: El investigador

Cada vez que se registre un candidato se va a aprobar una transacción.

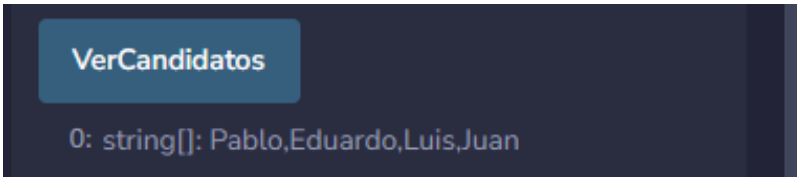


Figura 27 Cargar candidatos

Elaborado por: El investigador

En la figura 27 se muestran los candidatos que fueron cargados para la votación mediante la función VerCandidatos.

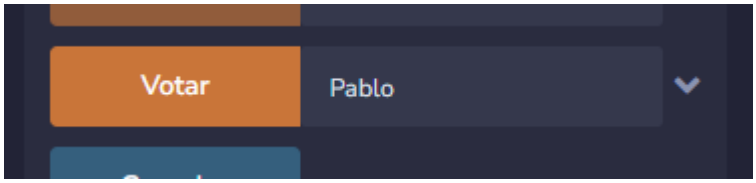


Figura 28 Votar por un candidato

Elaborado por: El investigador

Para votar por un candidato se debe poner el nombre del mismo y presionar el botón "Votar".

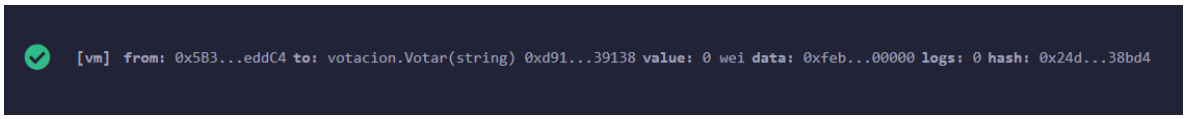


Figura 29 Transacción exitosa

Elaborado por: El investigador

Cada vez que se ejecute una función se registra una transacción en la Blockchain de Ethereum en este caso se registran en la máquina virtual. El Contrato Inteligente recibe y registra cada voto de manera segura y transparente en la red de Ethereum, asegurando que nadie pueda manipular o alterar los resultados.

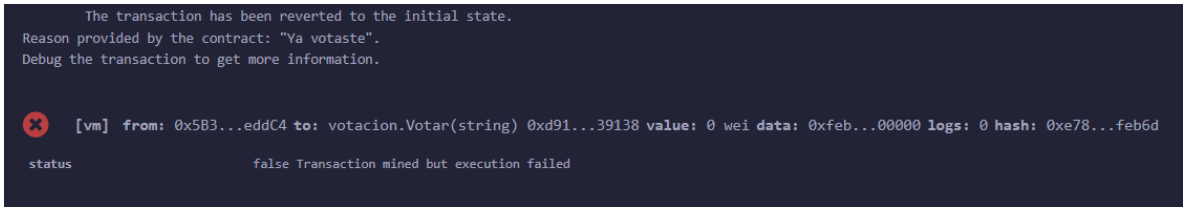


Figura 30 Votación fallida

Elaborado por: El investigador

El error que se genera en la Figura 30 es debido a que se volvió a intentar votar con la misma dirección para asegurar la integridad del proceso de votación. Esto se debe a que el Contrato Inteligente realiza un seguimiento de las direcciones de los votantes para evitar que una persona vote mas de una vez.

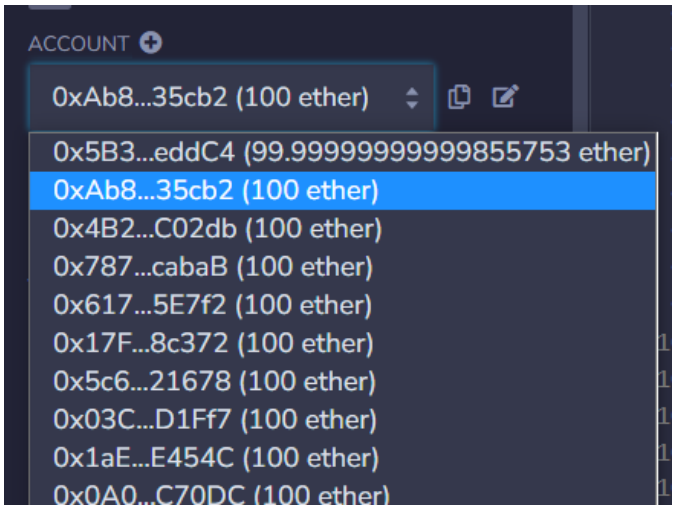


Figura 31 Cambiar la cuenta o dirección

Elaborado por: El investigador

Para poder realizar otra votación de prueba en Remix, se debe cambiar la cuenta o dirección como se muestra Figura 31.

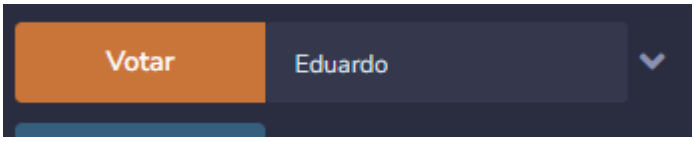


Figura 32 Voto con otra dirección

Elaborado por: El investigador

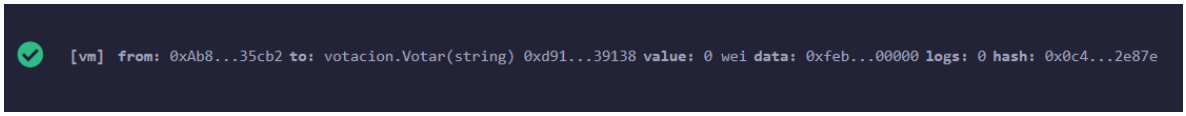


Figura 33 Transacción exitosa

Elaborado por: El investigador

Como se puede apreciar en la Figura 32 y Figura 33 la votación con otra cuenta o dirección nueva resulta exitosa. Y se genera otra transacción en la firma del Contrato Inteligente.

Pruebas de funcionamiento en una red de prueba de Ethereum

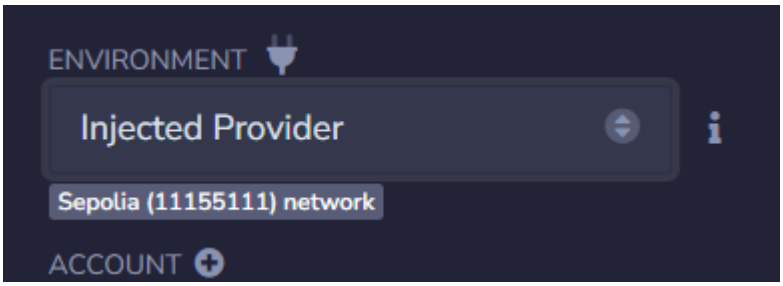


Figura 34 Cambiar el Environment

Elaborado por: El investigador

Para poder realizar las pruebas en una red de test de Ethereum se debe cambiar el campo “Environment” a “Injected Provider” así como se muestra en la Figura 34.

Figura 35 Solicitud conexión de Metamask

Elaborado por: El investigador

Esta conexión permitirá enlazar la billetera Metamask con el IDE Remix esto para probar las transacciones en Ethereum.

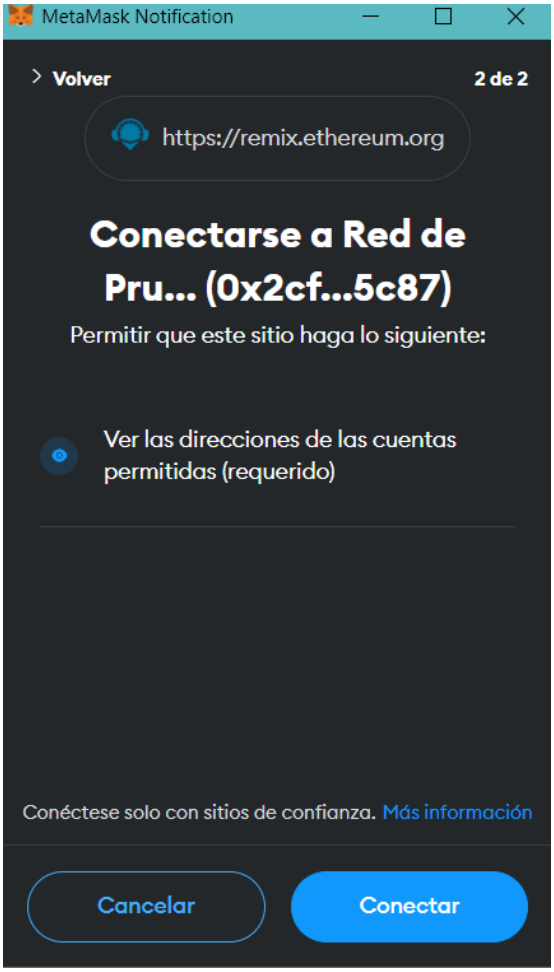


Figura 36 Permiso para Remix

Elaborado por: El investigador

Este permiso permitirá a Remix acceder a la cuenta, lo que quiere decir que se va a poder observar la dirección de la billetera en el entorno de Remix.

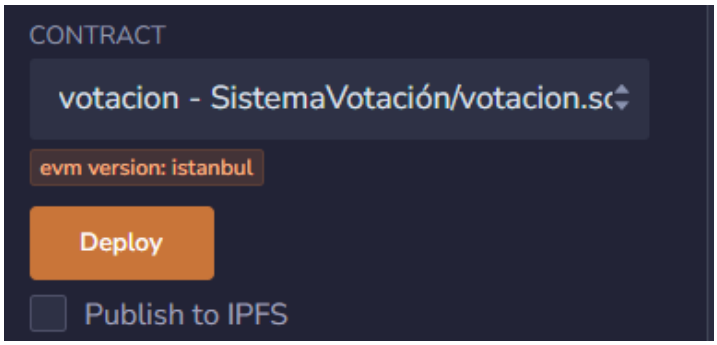


Figura 37 Desplegar el contrato

Elaborado por: El investigador

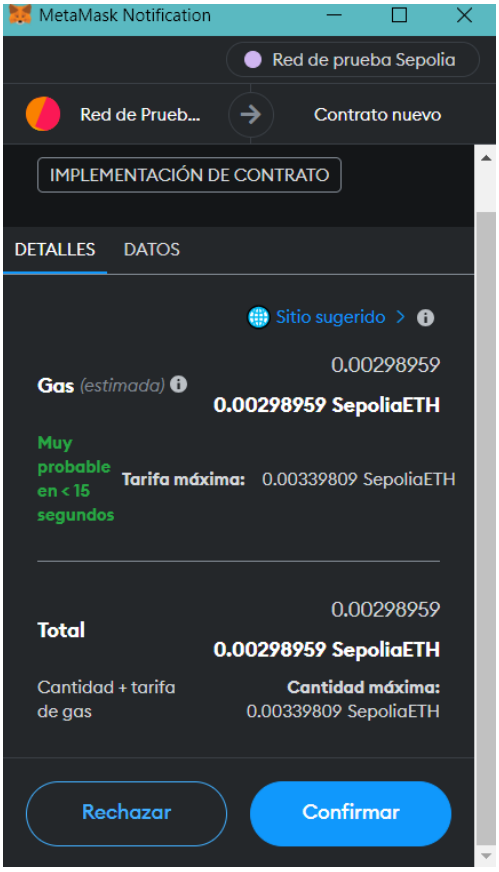


Figura 38 Transacción para desplegar el contrato

Elaborado por: El investigador

Una vez que se despliega el Contrato como se puede ver en la Figura 37, aparecerá una solicitud de Metamask para confirmar o rechazar la transacción, así como muestra la Figura 38. Esta transacción tiene un costo aproximado de 5 dólares estadounidenses esto transformado de la tarifa que se cobra en Ethereum para mantener la red.

Cabe mencionar que una entidad o un desarrollador puede crear una Blockchain privada especifica para el sistema que se requiera esto para evitar los costos de transacción.

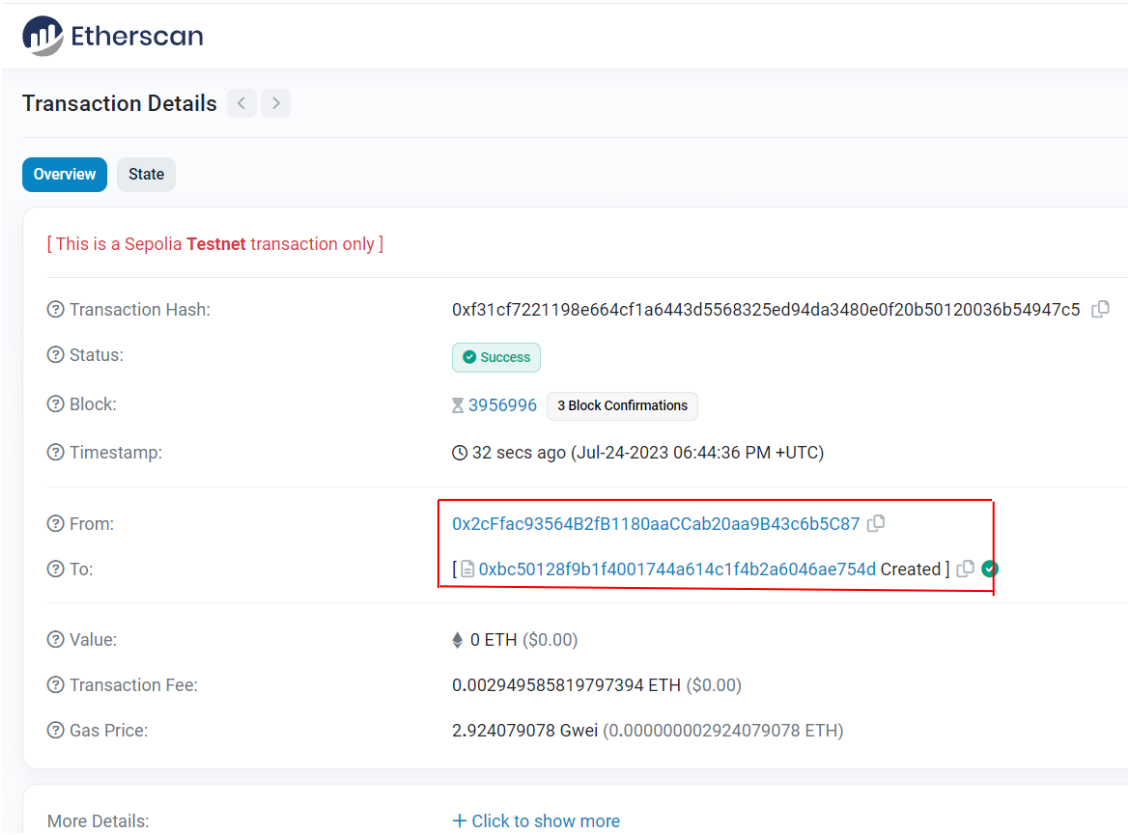


Figura 39 Detalles de la transacción

Elaborado por: El investigador

Con la herramienta Etherscan se puede revisar los detalles de la transacción que fue generada al desplegar el contrato en la red Ethereum. En la figura 39 se puede apreciar la dirección que desplegó el Contrato Inteligente y la firma de este.

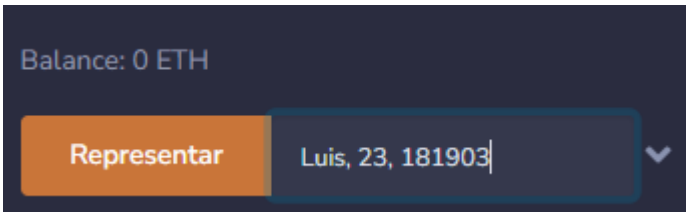


Figura 40 Cargar los candidatos

Elaborado por: El investigador

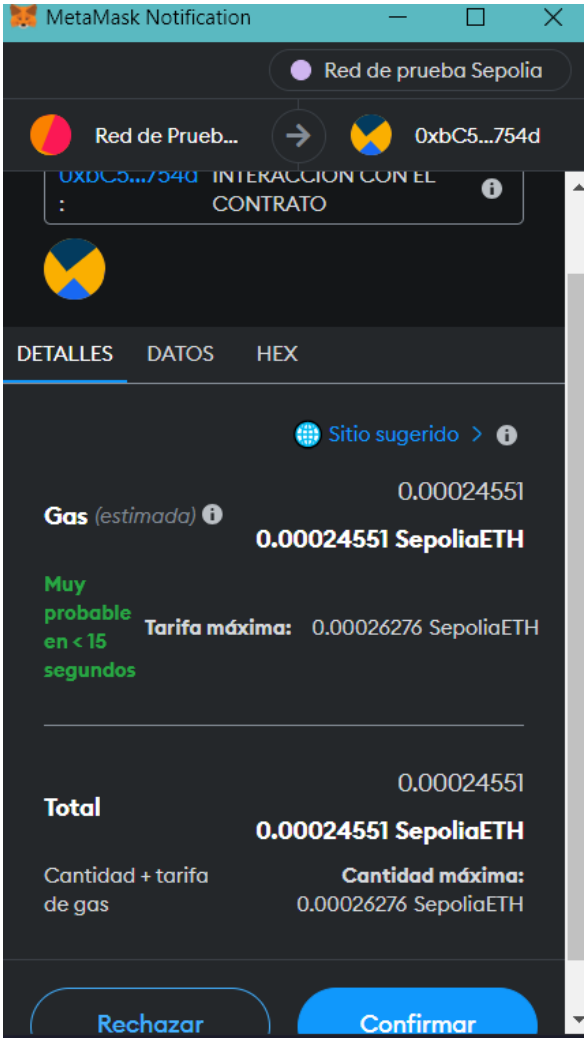


Figura 41 Transacción para cargar un candidato

Elaborado por: El investigador

Cada vez que se interactúa con el contrato dentro del sistema como se ve en la carga de del candidato en la Figura 40, va a solicitar una transacción a través de Metamask como muestra la Figura 41, la cual se puede confirmar o rechazar. Esta transacción tiene un costo aproximado de 45 centavos hay que mencionar que se está utilizando una red de test de Ethereum por lo cual el Ethereum que se está gastando en estas transacciones no tiene valor económico.

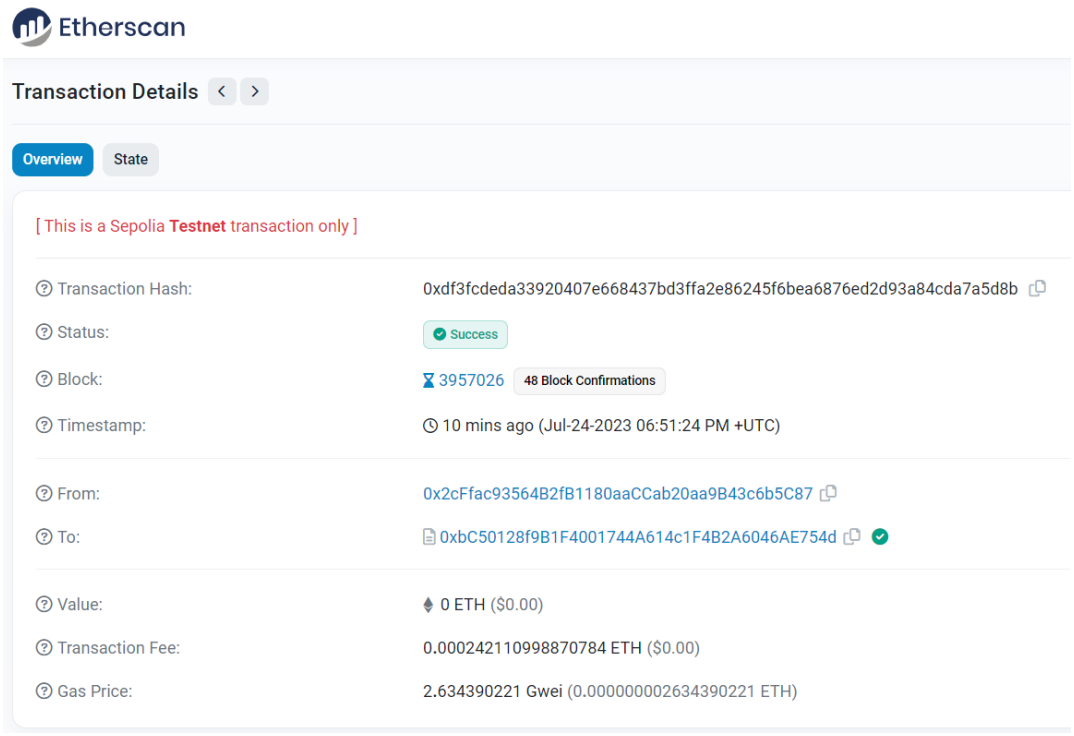


Figura 42 Detalles de la transacción

Elaborado por: El investigador

Se puede apreciar en la Figura 42 los detalles de la transacción que se generó por la interacción con el Contrato, también se puede observar que la transacción fue exitosa.

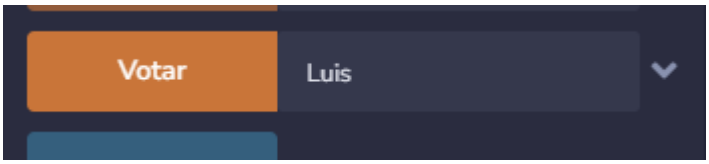


Figura 43 Votar por un candidato

Elaborado por: El investigador

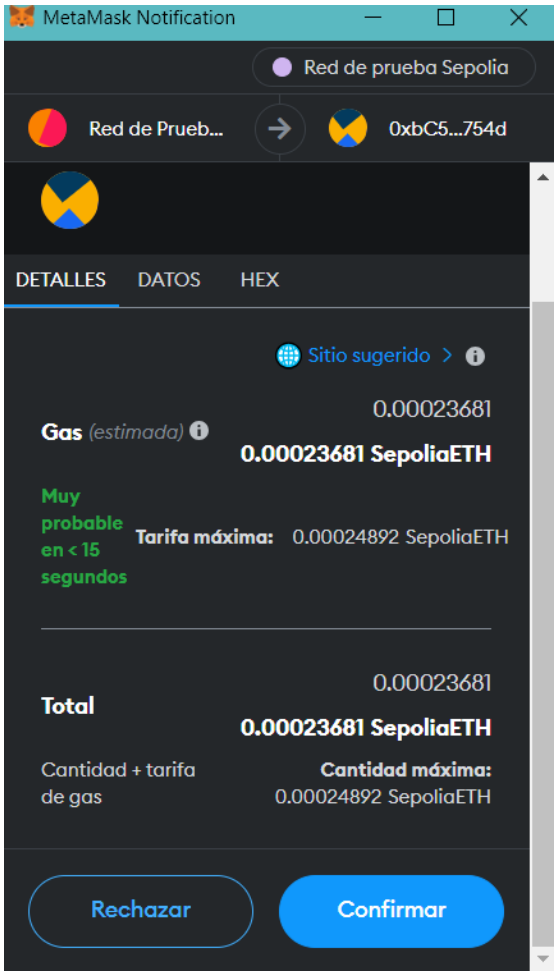


Figura 44 Transacción de la votación

Elaborado por: El investigador

Como se mencionó anteriormente cada vez que se interactúa con las reglas del Contrato se va a solicitar la confirmación de la transacción por medio de Metamask que es la

dirección que está conectada a Remix. Esta transacción tiene un costo aproximado de 44 centavos. Cabe destacar que este es la única transacción que realizaría el usuario.

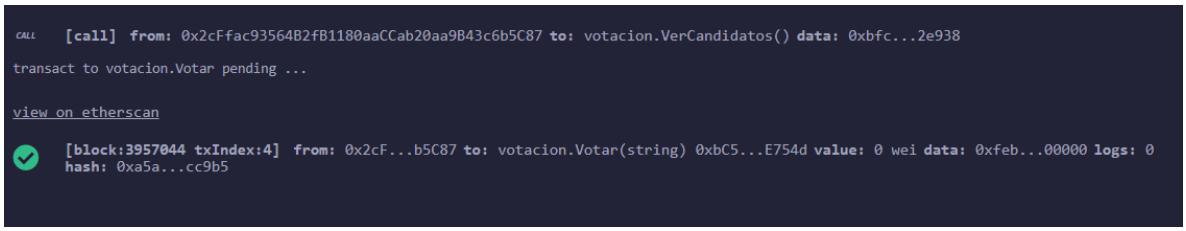


Figura 45 Transacción exitosa en Remix

Elaborado por: El investigador

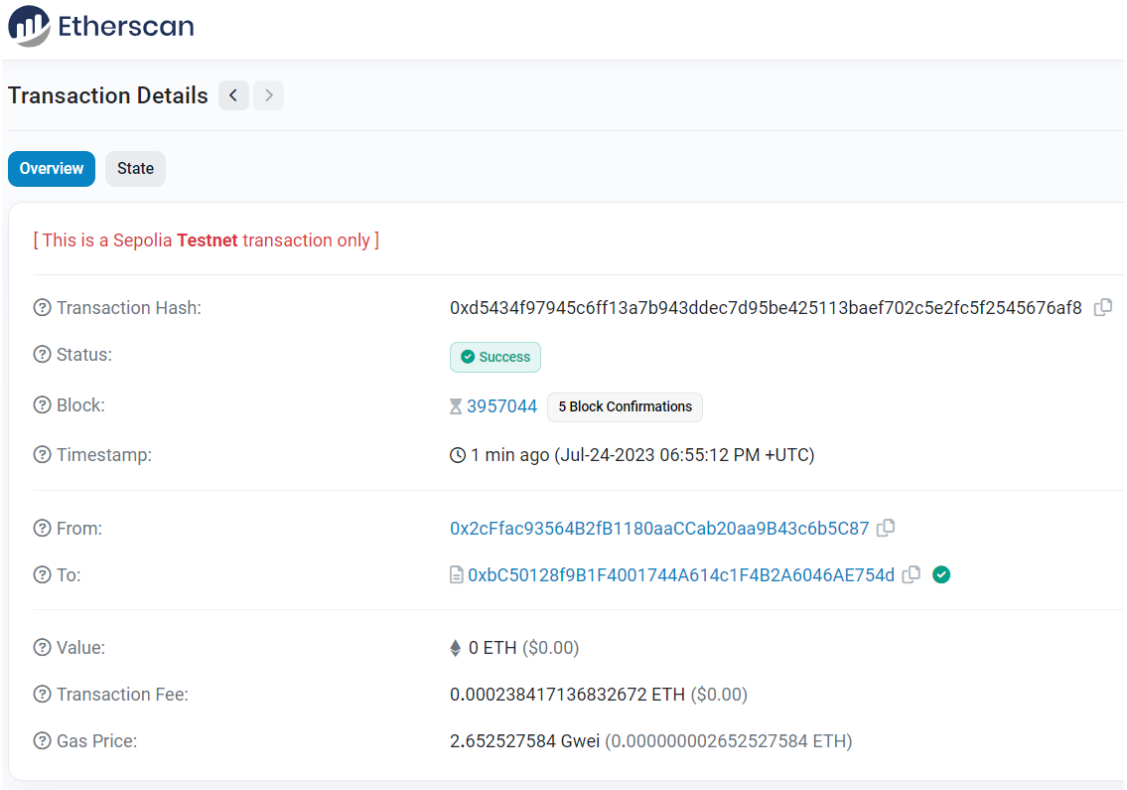


Figura 46 Detalles de la transacción en Etherscan

Elaborado por: El investigador

En la Figura 45 se puede observar que dentro de Remix la transacción fue exitosa, así también como se puede ver en la Figura 46 el éxito de la transacción en la red de Ethereum mediante Etherscan. Una vez que se hace el registro de la transacción los datos se guardan en la Blockchain, lo que hace que los datos sean inmutables.

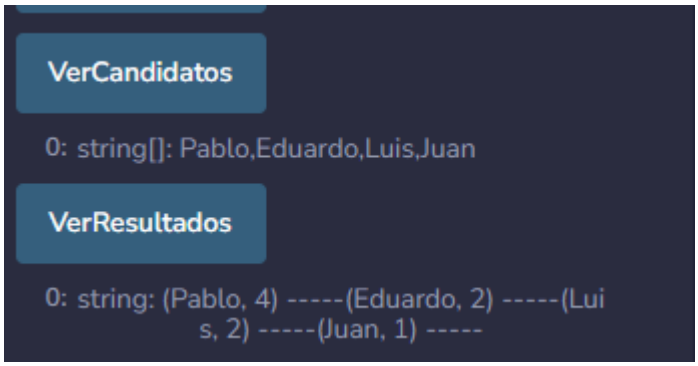


Figura 47 Ver candidatos y resultados

Elaborado por: El investigador

En la Figura 47 se muestra las funciones para ver los candidatos que se registraron para la votación, y de igual manera la función para ver los resultados de las votaciones. Una vez que finalizan las votaciones el Contrato Inteligente detiene la aceptación de nuevos votos y calcula los resultados. Estos resultados son públicos y transparentes, pero los votos individuales permanecen encriptados y anónimos.

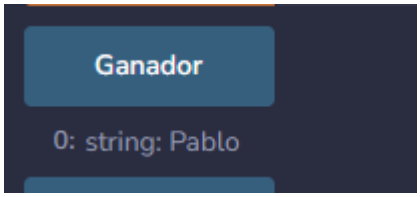


Figura 48 Ver el ganador

Elaborado por: El investigador

En la Figura 48 se muestra la función para ver el ganador de las votaciones que resulta ser Pablo con 4 votos. Esta función calcula y anuncia el ganador de manera automática e inmutable. Esto significa que nadie, incluido el creador del Contrato Inteligente pueda cambiar o manipular los resultados después de que se haya emitido el voto, lo que asegura la confiabilidad y confianza en el proceso de votación.

CAPÍTULO IV.- CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- Se logró una comprensión profunda de los riesgos que rodean la privacidad de los datos en el entorno de la Cadena de Bloques. La investigación reveló que, aunque la Cadena de Bloques proporciona una inmutabilidad y transparencia sin precedentes, también presenta desafíos en términos de privacidad debido a la visibilidad de los datos y a la dificultad de eliminación de información. Estos riesgos destacan la necesidad de estrategias sólidas para garantizar la confidencialidad de los datos sin comprometer las ventajas fundamentales de la tecnología.

- Las capacidades de los Contratos Inteligentes han demostrado que son una herramienta poderosa para el tratamiento seguro y eficiente de datos en la Cadena de Bloques. Su naturaleza autoejecutable, combinada con las características de seguridad incorporadas, garantiza que los datos sean compartidos de manera confiable y precisa. La eficiencia de los Contratos Inteligentes en la ejecución de tareas programadas abre un nuevo horizonte para la automatización de procesos sin sacrificar la seguridad de los datos.

- El diseño del modelo de la arquitectura de Cadena de Bloques basado en Contratos Inteligentes demostró ser una solución sólida para abordar los riesgos de privacidad identificados. La integración de Contratos Inteligentes en la arquitectura permitió un control más preciso sobre el acceso de datos, que aseguraba la integridad y confiabilidad de la información.

4.2 Recomendaciones:

- Se recomienda continuar monitoreando y actualizando las mejores prácticas de seguridad en el ámbito de la Cadena de Bloques y los Contratos Inteligentes. La

tecnología evoluciona rápidamente, y es importante mantenerse informado sobre las últimas tendencias y soluciones para garantizar la seguridad de los datos.

- Realizar pruebas del modelo de arquitectura propuesto en un entorno de prueba antes de su implementación en producción. Las pruebas permitirán identificar posibles vulnerabilidades y errores, lo que garantizará que el sistema sea robusto y confiable.
- Promover la colaboración y el intercambio de conocimientos entre la comunidad de desarrolladores de Cadena de Bloques. La colaboración entre profesionales permite compartir experiencias y lecciones aprendidas, lo que enriquece el desarrollo y la implementación de soluciones seguras y eficientes.

REFERENCIAS BIBLIOGRÁFICAS

[1] J. G. Horrach Armo, «Los acuerdos atributivos de jurisdicción en el ámbito de los smart contracts y la tecnología blockchain,» *Revista Electrónica de Estudios Internacionales,* 2021.

[2] T. Naúmenko y L. Fakhrutdínova, «LA TECNOLOGÍA BLOCKCHAIN EN AMÉRICA,» *Iberoamérica,* p. 51, 2019.

[3] R. Villacís Naranjo, «La legislación ecuatoriana en el uso de Blockchain,» *YURA: Relaciones Internacionales,* nº 18, p. 97, 2019.

[4] L. E. Rosero Correa, «Propuesta de una aplicación basada en la tecnología blockchain para el registro de títulos,» Quito, 2019.

[5] B. A. Castro Tapia, «Propuesta de una aplicación basada en la tecnología Blockchain y Smart Contracts para el resgistro de contratos de arrendamiento.,» Quito, 2021.

[6] J. A. Orrala Moreira, «Diseño de una guía para la elaboración de "smart contracts" en la red de Cardano para la empresa Naviera Panoil,» La Libertad, 2022.

[7] L. R. Álvarez Rojas, «Análisis de la tecnología blockchain, su entorno y su impacto en modelos de negocios,» Valparaíso, 2018.

[8] J. A. Triana Casallas, «Meta-Modelo de contratos inteligentes usando cadenas de bloques aplicado al sector público,» Oviedo, 2021.

[9] G. Baca Urbina , Introducción a la Seguridad Informática, México D.F.: Grupo Editorial Patria, 2016.

[10] M. Romero Castro , G. Figueroa Morán y D. Vera Navarrete , INTRODUCCION A LA SEGURIDAD INFORMÁTICA Y EL ANÁLISIS DE VULNERABILIDADES, 3Ciencias, 2018.

[11] J. Castro , «Blog Corponet | Todo sobre SAP Business One,» 28 Julio 2022. [En línea]. Available: https://blog.corponet.com/ciberseguridad-concepto-tipos-amenazas-estrategias. [Último acceso: 25 Marzo 2023].

[12] «Cloudflare,» [En línea]. Available: https://www.cloudflare.com/es-es/learning/privacy/what-is-data-privacy/. [Último acceso: 25 Marzo 2023].

[13] C. Dolader, J. Bel y J. Muñoz, «La blockchain: fundamentos, aplicaciones y relación con otras tecnologías disruptivas,» Cataluña.

[14] «El Derecho,» [En línea]. Available: https://elderecho.com/registro-distribuido-tokenizacion-blockchain-y-politicas-economicas-en-el-metaverso. [Último acceso: 25 Marzo 2023].

[15] «Bit2Me Academy,» [En línea]. Available: https://academy.bit2me.com/que-son-los-smart-contracts/. [Último acceso: 25 Marzo 2023].

[16] E. López, «REDES P2P Y CIBERDELINCUENCIA,» 2021.

[17] J. A. Lecuit, «La seguridad y la privacidad del blockchain, más allá de la tecnología y las criptomonedas,» Real Instituto Elcano, 2019.

[18] Y. M. Rendón, «Riesgos para la seguridad de la innovadora plataforma "Blockchain".,» Bogotá, 2019.

[19] A. Preukschat y C. Kuchkovsky, Blockchain: la revolución industrial de internet., Barcelona: Grupo Planeta, 2017.

[20] A. Taibo Escarramán, «Seguridad en la Blockchain de Ethereum: explotación y mitigación de vulnerabilidades modernas en Smart Contracts.,» Madrid, 2022.

[21] P. Martínez, «Identificación y control de riesgos en procesos validados con Blockchain.,» ISACA, 2020.

[22] G. Cardozo y P. Perdomo, «Comparación de plataformas para Smart Contracts basadas en Blockchain.,» Montevideo, 2020.

[23] V. Miranda Palacios, «Explorando la Blockchain de Ethereum y el desarrollo de Smart Contracts.,» 2018.

[24] Y. Calva Vega, M. Torres Villamarín, F. Cañizarez Galarza y J. Narvaéz Moncayo, «LOS CONTRATOS INTELIGENTES Y SU INCORPORACIÓN EN ORDENAMIENTO JURÍDICO,» Universidad Y Sociedad, 2022.

[25] S. X, «Santander X,» 21 Abril 2022. [En línea]. Available: https://www.santanderx.com/es/blog/que-son-los-smart-contracts.html. [Último acceso: 30 Mayo 2023].

[26] N. Rodríguez , «101 Blockchains,» 26 Agosto 2020. [En línea]. Available: https://101blockchains.com/es/que-es-un-contrato-inteligente/. [Último acceso: 30 Mayo 2023].

[27] «AR Wiki,» [En línea]. Available: https://aromatherapia.org/ventajas-y-desventajas-de-ethereum. [Último acceso: 7 Junio 2023].

[28] I. Flores, «Medium,» 5 Agosto 2018. [En línea]. Available: https://medium.com/inteligencia-log%C3%ADstica/hyperledger-fabric-resumen-y-conclusi%C3%B3n-bc43213a86d8. [Último acceso: 7 Junio 2023].

[29] «Bybit,» 27 Septiembre 2021. [En línea]. Available: https://learn.bybit.com/es/altcoins/what-is-cardano-ada-and-is-it-a-good-investment/. [Último acceso: 7 Junio 2023].

[30] «Invertligentes,» [En línea]. Available: https://inverligentes.com/altcoins/eos. [Último acceso: 7 Junio 2023].

[31] E. Bello, «IEBS,» 20 Febrero 2023. [En línea]. Available: https://www.iebschool.com/blog/solidity-lenguaje-programacion-ethereum-tecnologia/. [Último acceso: 7 Junio 2023].

[32] J. L. Pascal, «bit2me Academy,» 26 Mayo 2022. [En línea]. Available: https://academy.bit2me.com/top-5-de-lenguajes-de-programacion-de-smart-

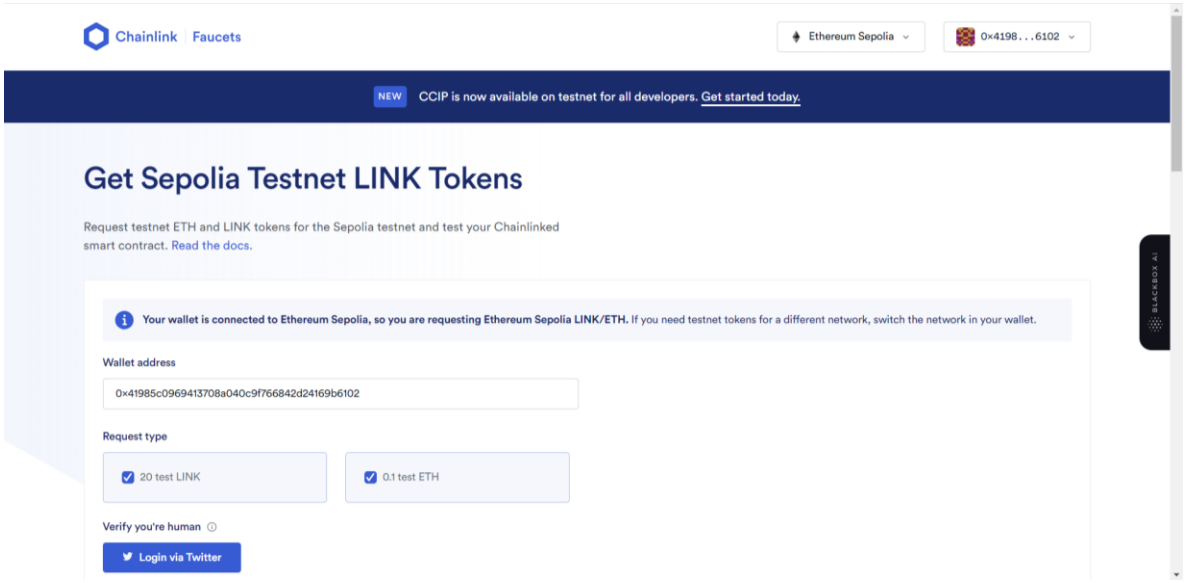
contracts/. [Último acceso: 23 Junio 2023].

ANEXOS

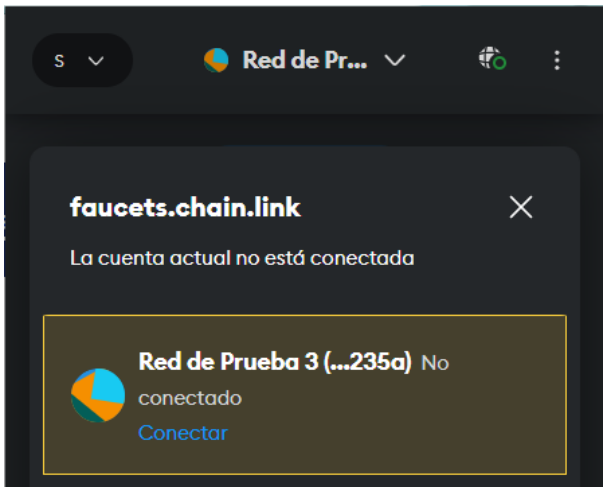
Anexo 1: Obtener Ethereum (Criptomoneda) de prueba para el desarrollo.

https://faucets.chain.link/

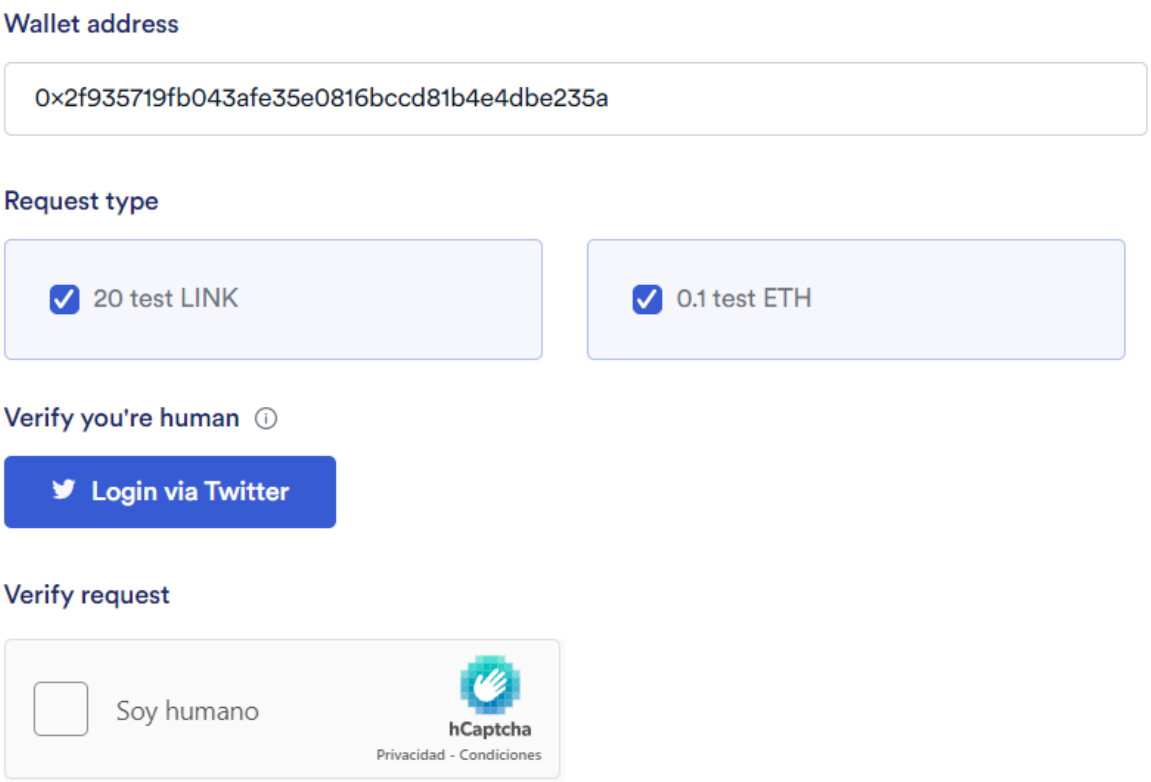
Url del sitio para conseguir Ethereum de prueba para desarrollo.



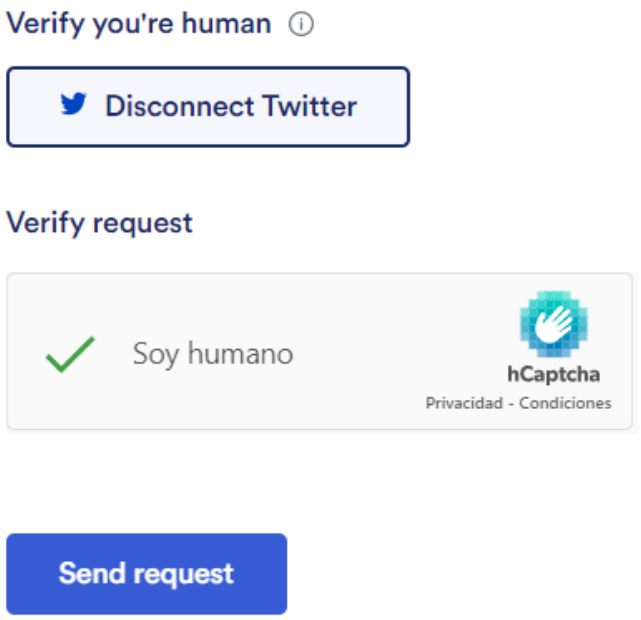
Para obtener Ethereum de prueba hay que seleccionar una red de test de Ethereum que en este caso es Sepolia.



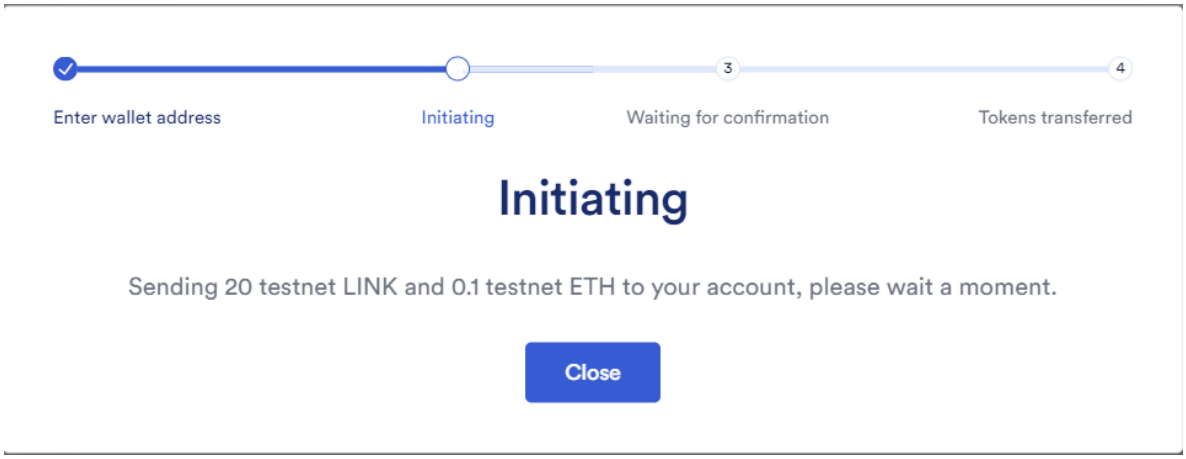
Primero se debe conectar una billetera digital de Metamask que esté en la red de Sepolia al sitio.



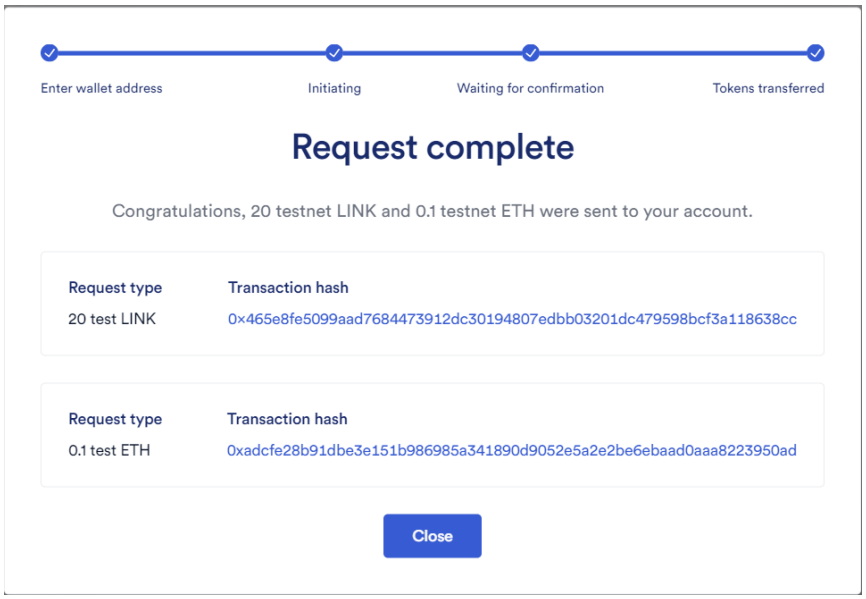
Completar los pasos de verificación.



Una vez completada la verificación se envía la solicitud.



Se inicializa la solicitud.



Luego de que se completó la solicitud se confirma el envío del Ethereum de prueba.

Una vez recibido el Ethereum ya se lo puede utilizar para el desarrollo.