

**Escuela de Ingeniería
Maestría en Seguridad Informática**

Aspectos generales sobre la adopción de la tecnología Blockchain en MiPyMes mexicanas: activos virtuales como medio de pago e implementación de contratos inteligentes

TESIS
QUE PARA OBTENER EL TITULO DE MAESTRO EN
SEGURIDAD INFORMATICA

P R E S E N T A
GERARDO ALBERTO CATAÑO CAÑIZALES

Ciudad de México

Febrero de 2022

Resumen

La tecnología Blockchain representa un cambio de paradigma en la computación como la conocemos hoy en día. En los próximos años, servicios tecnológicos tradicionales evolucionarán y nuevas soluciones surgirán siguiendo los lineamientos de la Web 3.0, donde adoptarán un nuevo enfoque global, descentralizado, distribuido, transparente y libre.

Esta investigación pretende ser un punto de partida accesible para emprendedores y microempresarios dispuestos a innovar dentro de sus negocios, al incorporar tecnologías Blockchain y obtener algunos de sus beneficios inmediatos.

Mediante la explicación de los fundamentos alrededor de los conceptos de Blockchain, activos virtuales y contratos inteligentes, esta investigación se centra en instruir al lector de una manera sencilla y asistida en cómo integrar un método de pago con criptomonedas para sus actividades comerciales y en cómo propiciar un nuevo tipo de interacción con sus clientes a través de la implementación de un contrato inteligente.

Ambos mecanismos son expuestos de manera que el lector pueda realizar paso a paso su implementación, y tienen la finalidad de permitir al comerciante ofrecer sus bienes y/o servicios de una manera vanguardista y atractiva para la ciudadanía digital.

Palabras Clave: Blockchain, Bitcoin, Criptomoneda, Ethereum, Contrato Inteligente, Aplicación Descentralizada.

Maestría en Seguridad Informática

UNIVERSIDAD
INTERNACIONAL
DE LA RIOJA



Tabla de Contenido

1	Introducción	1
1.1	Antecedentes	1
1.2	Objetivo	1
1.2.1	Objetivo General	1
1.2.2	Objetivos Específicos.....	2
1.3	Justificación	2
1.4	Metodología.....	3
1.5	Organización del Documento	3
2	La tecnología Blockchain	5
2.1	¿Qué es Blockchain?	5
2.2	Origen de la tecnología	6
2.3	¿Cómo funciona?.....	7
2.4	Sustento criptográfico	8
2.5	Componentes y características	9
2.6	Blockchain público vs privado	11
2.6.1	Blockchain público	11
2.6.2	Blockchain privado	11
2.7	Casos de uso	12
3	Activos virtuales.....	14
3.1	Descentralización	14
3.2	¿Qué son los activos virtuales o criptomonedas?.....	15
3.3	¿Qué es Bitcoin?	17
3.4	Wallets	18
3.5	Minado de criptomonedas	22
3.6	Transacciones	24
4	Contratos inteligentes.....	27
4.1	¿Qué son los contratos inteligentes o smart contracts?	27
4.2	Plataforma Ethereum.....	27
4.3	Aplicaciones descentralizadas	29
4.4	Modelo de pago: Gas.....	31
4.5	Transacciones en Ethereum	32

4.6 Desarrollo en Ethereum.....	33
4.6.1 Ethereum Virtual Machine	33
4.6.2 Lenguaje de programación Solidity.....	33
4.6.3 Redes en Ethereum.....	34
5 Propuesta de implementación	35
5.1 Objetivos de la propuesta	35
5.2 Alcances y limitaciones	35
5.3 Adopción de mecanismo de pago con criptomonedas.....	36
5.3.1 Creación de cuenta de usuario en el exchange Bitso	36
5.3.2 ¿Cómo recibir un pago con bitcoin?	43
5.3.2.1 Dirección bitcoin de autoventa.....	44
5.3.2.2 Dirección bitcoin dinámica	46
5.3.3 ¿Cómo realizar un pago con bitcoin?	49
5.3.4 ¿Cómo recibir y realizar un pago con otra criptomoneda?	55
5.4 Prueba de concepto para contrato inteligente	57
5.4.1 Configuración de Wallet MetaMask.....	57
5.4.2 Entorno de desarrollo Ethereum Remix IDE.....	66
5.4.3 Definición de caso de uso para Smart Contract	69
5.4.4 Codificación y compilación del Smart Contract	70
5.4.5 Despliegue del Smart Contract	73
5.4.6 Interacción con el Smart Contract.....	76
6 Conclusiones.....	84
6.1 Conclusiones	84
6.2 Recomendaciones	85
6.3 Trabajos futuros	85
Referencias.....	87
Bibliografía.....	89
Apéndice A - Acrónimos	92
Apéndice B - Glosario de Términos	93
Apéndice C - Índice de figuras	95
Apéndice D - Otros recursos	98

1 Introducción

1.1 Antecedentes

La descentralización de servicios y aplicaciones como principal motor la Web 3.0 suponen un área de oportunidad que emprendedores y micro empresas deben empezar a aprovechar para ser más competitivos, mantenerse a la vanguardia tecnológica y ofrecer servicios atractivos para la ciudadanía digital.

Actualmente, la implementación de tecnologías Blockchain sigue siendo una tendencia global que presenta una sólida alternativa a los servicios financieros tradicionales. Organizaciones globales, empresas nacionales y multinacionales como Visa, Mastercard, Paypal, Mercado Libre, Tigres U.A.N.L., Grupo Elektra, Federación Mexicana de Fútbol Asociación, A. C. entre otras, recientemente las han ido adaptando de diferentes maneras a sus modelos de negocio particulares, a través de alianzas estratégicas y de grandes inversiones de capital.

Por otro lado, las altas comisiones, dependencia de terceros para efectuar cualquier tipo de operación financiera y su no disponibilidad 24/7/365 (24 horas al día/7 días a la semana/365 días al año) son problemas comunes con los que puede lidiar frecuentemente una MiPyMe; no obstante, estos pueden ser aminorados o erradicados con relativa facilidad a través de la implementación de tecnologías Blockchain, como lo abordaremos en la presente investigación.

1.2 Objetivo

1.2.1 Objetivo General

Guiar metódicamente al mayor número de emprendedores acerca de los diferentes beneficios que se pueden obtener de la tecnologías Blockchain, como las

características intrínsecas de Seguridad de la Información que estas otorgan, la eliminación de intermediarios durante los procesos clave del negocio con sus consecuentes ahorros, y brindar una ventaja competitiva ante sus pares a través de la adopción temprana de esta tecnología de vanguardia mediante una inversión modesta, principalmente en conocimiento.

1.2.2 Objetivos Específicos

- Exposición de estudio a alto nivel del ecosistema Blockchain, presentando sus principales características, elementos que lo componen, funcionamiento, beneficios que otorga y ejemplos de casos de uso.
- Elaboración de guía funcional simplificada para la implementación de mecanismo de pago con criptomonedas orientada hacia emprendedores y MiPyMes, con el apoyo de los servicios ofrecidos por una FinTech (empresa de tecnología financiera) mexicana.
- Desarrollo de guía técnica y prueba de concepto de un contrato inteligente, que sirva de ejemplo o plantilla para la innovación de nuevos casos de uso orientados hacia los negocios particulares de emprendedores y MiPyMes mexicanas.

1.3 Justificación

A pesar de no tratarse de un tópico nuevo, se hace evidente el desconocimiento sobre el tema y el rezago sobre su adopción en México y en países latinoamericanos (con sus contadas excepciones), como históricamente ha sucedido con la implementación de nuevas tecnologías para esta región. Por este motivo, este documento pretende ser una guía simplificada acerca de los primeros pasos para la adopción de las tecnologías Blockchain y de los diferentes beneficios que se pueden obtener de ellas.

en el día a día de las operaciones de una MiPyMe en México.

Adicionalmente, el impacto económico que su adopción podría generar en un negocio puede ser importante, ya que no se necesitan de grandes inversiones tecnológicas o de capital, dado que actualmente existen plataformas de tecnología financiera y herramientas de Software libre que facilitan su implementación.

1.4 Metodología

Esta consistirá en la exposición de los conceptos generales de Blockchain y su relación entre sí. Posteriormente y abarcando aspectos funcionales y técnicos, se llevará a cabo la descripción detallada del como aceptar activos virtuales (criptomonedas) como forma de pago a cambio de los bienes y/o servicios ofrecidos por una MiPyMe y de la implementación de un contrato inteligente (smart contract) para un caso de uso genérico, el cual pueda resultar útil para la mayoría de dichas entidades.

1.5 Organización del Documento

La presente investigación se encuentra organizada en seis capítulos.

En el capítulo 1 se proporciona el contexto global de la investigación, los antecedentes, objetivos, justificación y la metodología que se siguió para la misma.

El capítulo 2 presenta las generalidades necesarias para la comprensión de la tecnología Blockchain y su ecosistema.

En los capítulos 3 y 4, se abordan los conceptos generales de los activos virtuales y

las nociones básicas de los contratos digitales, respectivamente.

En el capítulo 5 se expone el trabajo realizado para llevar a cabo la propuesta de adopción del mecanismo de pago con criptomonedas y la implementación de la prueba de concepto para contrato inteligente, abarcando los aspectos técnicos y funcionales necesarios para su desarrollo.

Finalmente, en el capítulo 6 se presenta una recapitulación de los puntos más relevantes de la investigación, conclusiones, recomendaciones y líneas de investigación para trabajos futuros.

2 La tecnología Blockchain

2.1 ¿Qué es Blockchain?

De acuerdo con Gupta (2017, p. 3), Blockchain se define como un libro contable digital, compartido y distribuido que facilita el proceso de registro de transacciones y el seguimiento de activos en una red comercial.

Blockchain es una red global igual a igual (peer to peer) de computadoras, nativamente orientada a objetos, donde cualquiera de sus participantes puede acceder de forma segura a datos y puede ejecutar código transaccional.

En una analogía simple, Blockchain puede visualizarse como una base de datos distribuida, formada por bloques de información.

Cada uno de los bloques que lo conforman contiene transacciones, principalmente financieras, y debido al mecanismo criptográfico a través del cual fueron almacenadas se hace casi imposible su manipulación o falsificación, una vez que estos fueron añadidos al Blockchain.

No existe una organización individual, compañía o gobierno que pueda tomar el control de un Blockchain o que pueda restringirlo total o parcialmente; tampoco tiene un punto central de fallo y es también muy resistente a hackeos dada su naturaleza distribuida.

Un Blockchain es fácilmente accesible desde cualquier punto donde se cuente con Internet, a través de una amplia variedad de clientes y tecnologías.

Finalmente, cada una de las transacciones en un Blockchain puede ser verificada por cualquier participante de la red, y además puede ser rastreada desde sus orígenes.

2.2 Origen de la tecnología

Existen varios antecedentes aislados que han ido dando forma a lo que hoy son las tecnologías Blockchain.

En 1991, Stuart Habber y Scott Stornetta describieron por primera vez las nociones de una cadena de bloques asegurada criptográficamente.

Por otro lado, el primer trabajo reconocido acerca de una divisa digital descentralizada que utilizaba una tecnología similar fue realizado por Nick Szabo en 1998.

No obstante, no fue hasta el 31 de octubre de 2008 que los conceptos de Bitcoin y Blockchain fueron mencionados por primera vez en el *whitepaper “Bitcoin: A Peer-to-Peer Electronic Cash System”* escrito por Satoshi Nakamoto, de quien al día de hoy no se conoce su verdadera identidad, o si en realidad se trata de un grupo de personas.

El 8 de enero de 2009 el bloque génesis de Bitcoin fue lanzado por Satoshi Nakamoto, creando con ello la primera divisa descentralizada.

Desde el año 2011 más criptodivisas han sido lanzadas, muchas de ellas han sido clones de infraestructuras Blockchain existentes, pero otros más se han ido desarrollando como proyectos especializados hacia la resolución de problemas más específicos.

A partir del año 2014, muchas compañías globales importantes han implementado la adopción y utilización de Bitcoin como parte de sus nuevas estrategias digitales de negocio y en enero de ese mismo año, surgió la primera plataforma mexicana de compra/venta de activos virtuales con operaciones en América Latina: Bitso S.A.P.I. de C.V.

2.3 ¿Cómo funciona?

La manera en la que Blockchain opera comienza con una persona o entidad realizando una o varias transacciones, las cuales por lo general envían información en forma de contrato y dependiendo del tipo de implementación utilizada, también podría involucrar el envío de criptomonedas de una cuenta a otra.

Cada transacción es enviada a una red de computadoras peer to peer distribuida por todo el mundo, donde a cada computadora se le conoce como nodo, y cada nodo contiene una copia completa de los datos existentes en el Blockchain.

La transacción se ejecuta y se valida a través de contratos previamente compartidos, lo cual asegura que todos los nodos estén ejecutando las mismas reglas.

Una vez que la transacción es ejecutada, el resultado es agregado al Blockchain.

Cabe señalar que estas acciones se realizan dentro de cada nodo y para poder comprometer una transacción, sería necesario comprometer a todos los nodos de la cadena.

Por otro lado, las transacciones en Blockchain son atómicas, lo cual significa que todos los pasos en una operación deben ejecutarse por completo o en caso contrario, ninguno de ellos.

También, se debe tomar en cuenta que las transacciones son ejecutadas independientemente unas de otras, por lo tanto no pueden interactuar directamente entre sí y además hay que asegurarse de que su ejecución sea en el orden correcto.

En Blockchain, toda la información y el código dentro de un objeto son permanentes. Se pueden crear nuevas transacciones y hacer consultas posteriores sobre su historial, sin embargo no se podrán hacer actualizaciones ni tampoco borrarlas; esto para mantener la consistencia criptográfica de la cadena de bloques, lo cual se

explicará en el siguiente punto.

2.4 Sustento criptográfico

Uno de los elementos más importantes para cualquier tecnología Blockchain radica en el concepto de hashing.

El hashing consiste en la ejecución de un algoritmo matemático que puede tomar una entrada de cualquier longitud, procesarla y generar como resultado una salida de una longitud determinada, conocido como hash.

El hashing es una función unidireccional, lo que significa que siempre regresará el mismo resultado para una entrada específica, sin embargo nunca podrá regenerar los datos de entrada a partir del resultado de la ejecución del algoritmo.

Las funciones hash fueron construidas para ser rápidas de calcular y bastante baratas, hablando en un contexto computacional.

También, son frecuentemente utilizadas para realizar verificación de integridad en archivos, en donde primeramente se calcula el hash del archivo y luego se compara con el hash esperado; si ambos valores coinciden el archivo está íntegro y es consistente, pero en caso contrario el archivo fue modificado. Debido a que es casi imposible encontrar dos archivos diferentes con el mismo hash o bien, modificar un archivo de tal manera que el hash siga siendo el mismo, esta verificación de integridad es muy eficaz.

Existen diferentes tipos de funciones hash pero todas ellas poseen propiedades muy similares; los más utilizados son bien conocidos por la comunidad y están disponibles para todo el público.

El algoritmo SHA-256, creado por la NSA (National Security Agency) de los Estados Unidos, es uno de los más utilizados dentro de tecnologías Blockchain y este por lo general se aplica a todas las transacciones en un bloque.

Otro concepto criptográfico relacionado con el hash es el árbol de Merkle, el cual básicamente es un hash de hashes. Su funcionamiento consiste en agrupar por pares los hashes de cada transacción y genera un nuevo hash; este proceso se repite recursivamente hasta obtener un root hash, el cual será el identificador del bloque.

Para verificar si hubo modificaciones en el árbol de Merkle, primero se revisa si el root hash cambió y luego se va explorando el árbol hacia abajo para identificar la transacción donde tuvo lugar ese cambio.

2.5 Componentes y características

Como se pudo inferir de su nombre, un Blockchain es una lista o cadena de bloques donde por lo general, cada bloque está compuesto principalmente por el número de bloque, el mensaje, su hash, el nonce (number that can be only used once), la marca de tiempo (timestamp) y el hash del bloque previo.

Todo bloque está formado por información y el hash que resulta de esta, por lo que si se llega a modificar cualquier dato en el bloque, el hash cambiará y por consecuencia el bloque sería inválido.

El número de bloque es el orden secuencial que este tiene dentro de la cadena de bloques, en tanto que el timestamp corresponderá a su fecha de creación.

En un bloque también se encuentra el nonce, el cual se utiliza como una entrada del algoritmo de hashing, y el cual resultará en un valor predefinido para la primera parte del hash, como por ejemplo un prefijo de “n” cantidad de ceros.

Dado que no es posible predecir un nonce, la creación del hash con estas características especiales puede ser considerada como una prueba de trabajo (Proof of Work) del nodo que esté tratando de generarlo. De esta forma, el nodo tendrá que ejecutar el algoritmo tantas veces sea necesario hasta descubrir que nonce utilizar para cumplir con los requisitos del hash. A este trabajo se le conoce como "minar el bloque".

Otro componente fundamental en un bloque es la referencia hash del bloque anterior, puesto que si existe alguna alteración en uno de los bloques, esto invalidaría todos los bloques subsecuentes de la cadena.

Por otra parte, en Blockchain una cadena de bloques se distribuye a través de un vasto número de nodos, esto quiere decir que una cadena de bloques existe en múltiples ubicaciones y dependiendo de la implementación, estas podrían ser millones de réplicas.

Con todos estos nodos vigilantes se puede averiguar fácilmente si una cadena fue alterada o minada de nueva cuenta, ya que sus hashes serían diferentes y por lo tanto al momento de ser verificados dentro de la secuencia, estos serían rechazados y removidos del Blockchain.

Teóricamente, sería posible alterar la historia de una cadena minando todos los bloques existentes después del bloque modificado, recreando todos sus hashes a través de la respectiva Proof of Work necesaria para descubrir cada nonce y posteriormente distribuyendo la solución hacia los otros nodos, sin embargo, esto requeriría de disponer del mismo poder computacional que el resto de los nodos de la red Blockchain juntos para poder lograr que la nueva cadena fuera aceptada.

Debido a que se requiere tanta potencia informática para hacer esto posible, realmente es una mejor alternativa utilizar estos recursos para minar bloques de manera legal.

2.6 Blockchain público vs privado

2.6.1 Blockchain público

Las implementaciones de Blockchain pueden ser tanto del tipo público como del privado.

En un Blockchain público, no existe una entidad central que controle la infraestructura y cualquiera que tenga conexión a Internet podrá accederlo.

Dado que sus nodos se encuentran altamente distribuidos por toda la red y cada uno de ellos contiene una copia completa del Blockchain, tampoco existe un único punto de fallo que se pueda explotar ante ataques informáticos.

Una de las principales fortalezas de un Blockchain público es el respaldo de su comunidad y la democracia que impera en su organización, ya que por lo general sus participantes activos pueden votar en las decisiones que lo afectan, como por ejemplo, el cómo manejar una bifurcación (fork).

En un Blockchain público, todas las transacciones son también públicas y permanecerán anónimas a menos que se asocie directamente una dirección Blockchain con la identidad de una de las entidades involucradas en una transacción.

Este último punto fue crítico para grandes compañías, ya que la inconformidad de tener información sensible expuesta hacia cualquiera persona, las motivó a abordar soluciones más orientadas a sus propios intereses, en este caso los Blockchain privados.

2.6.2 Blockchain privado

Existe controversia en si un Blockchain privado puede considerarse una solución Blockchain completa, ya que esta presenta características más semejantes a las de

una base de datos tradicional.

Un Blockchain privado es normalmente creado por un consorcio de compañías involucradas en la resolución de un escenario de negocios en común.

En este, solo entidades conocidas pueden participar, debido a que los propietarios tienen el control de toda la infraestructura y pueden decidir quién puede o no acceder la implementación.

Esta característica representa una ventaja, ya que de esta manera la privacidad de la información puede ser controlada por el consorcio, lo cual es necesario cuando existen de por medio medidas normativas que hay que acatar en industrias estrictamente reguladas, como el sector salud y el sector financiero.

Otra ventaja de este tipo de implementación es que esta tiene un mejor desempeño ya que la red es más pequeña, con menos nodos y por consecuencia, existe menor competencia en el minado de nuevos bloques.

No obstante, estas características también representan una gran desventaja, ya que la infraestructura sería más fácil de comprometer al no contar con nodos altamente distribuidos, como es el caso de su contraparte pública.

2.7 Casos de uso

Existen varios casos de uso de Blockchain que cruzan una amplia variedad de industrias. A continuación se describirán algunos ejemplos de sus usos actuales más comunes.

Alrededor del mundo y sobre todo en países en vías de desarrollo, existen millones de personas que no tienen pruebas suficientes para poder establecer su propia

identidad y por ende, no son elegibles para recibir servicios básicos, públicos o financieros. Por este motivo, organizaciones sin fines de lucro se encuentran desarrollando iniciativas Blockchain para la emisión de certificados de nacimiento que son autenticados digitalmente, son fáciles de verificar, no se pueden falsificar y son accesibles para cualquier persona.

Por otro lado, en el área de gestión de la cadena de suministro (supply management), Blockchain puede mantener información detallada acerca de la procedencia de cada uno de los componentes necesarios para la manufactura o ensamblaje de un producto final, como el estado actual de una pieza, fecha de producción, lote, entre otros, y esta su vez puede ser compartida en todo momento por entidades involucradas en el proceso como fabricantes, distribuidores, aseguradoras y reguladores gubernamentales.

En el sector salud, el Blockchain ha sido utilizado en implementaciones que guardan el registro médico histórico de un paciente de manera segura y confiable; además, este puede ser compartido entre las diferentes entidades involucradas como médicos, hospitales y aseguradoras, pudiendo establecer diferentes niveles de visibilidad en su contenido o limitarlo por un periodo de tiempo determinado, según se requiera.

Otra área de aplicación que va en ascenso es la del Internet de las Cosas (IoT), donde los diferentes sensores y dispositivos electrónicos generan y comparten información unos con otros a través de Internet. Con el uso de Blockchain, los dispositivos pueden garantizar transacciones seguras y automatización de procesos a través de contratos inteligentes (smart contracts), además de proporcionar trazabilidad y no repudio en sus operaciones.

Finalmente, el uso más extendido de la tecnología Blockchain es el de las divisas digitales o criptomonedas. Este tema se abordará con mayor profundidad en el siguiente capítulo.

3 Activos virtuales

3.1 Descentralización

En un sistema centralizado, como en un banco, este es el único responsable por el procesamiento de todas las transacciones financieras y por mantener actualizado su registro y el del balance de las cuentas de cada uno de sus clientes. Por este motivo, se debe de tener una completa confianza en la institución.

Uno de los inconvenientes que estas entidades financieras suelen presentar es la lentitud de sus transacciones, ya que estas pueden llegar a tardar incluso días cuando se realizan de una institución hacia otra, y especialmente con transferencias al extranjero. Este último escenario también implica el cobro de altas comisiones y por consecuencia, desalienta la realización de micro transacciones.

Otro inconveniente, es que durante un ataque informático o incidente grave dentro de la entidad financiera, existe una alta probabilidad de que sus servicios presenten intermitencia o una indisponibilidad total o parcial durante el periodo de tiempo que tome la resolución del problema, esto debido al menor número de nodos consecuencia de la centralización de sus sistemas.

La descentralización en cambio, es un concepto clave y beneficio del Blockchain, donde no existe la necesidad de un tercero de confianza o un intermediario que deba validar las transacciones; en su lugar, se utiliza un mecanismo de consenso para acordar su validez (Bashir, 2017, p. 57).

Esto significa que en un sistema descentralizado como Blockchain, no se tiene que confiar en algún actor en particular de la red porque ninguno de sus participantes puede controlarla por completo. En su lugar, se puede confiar en el código y en los algoritmos ejecutándose en los nodos de esta red.

Las transacciones anónimas en Blockchain pueden ser tan confiables como las de

cualquier banco, lo cual arroja muchas posibilidades sobre todo cuando se trata de transacciones globales; en esencia, se pueden realizar transferencias directamente entre los involucrados, con comisiones muy bajas y de una manera casi instantánea a través de sencillas aplicaciones móviles.

3.2 ¿Qué son los activos virtuales o criptomonedas?

Descrito en el whitepaper de Satoshi Nakamoto en 2008, Bitcoin nace como la primera implementación exitosa de Blockchain y también como un mecanismo electrónico de pago: la criptomoneda.

Las divisas digitales, criptodivisas, activos virtuales o más comúnmente criptomonedas, son un medio de intercambio digital, cifrado y descentralizado, que se vale de la infraestructura Blockchain existente para la realización de pagos electrónicos destinados a la adquisición de bienes y servicios, tanto digitales como físicos. Estas deben su nombre a la alta dependencia que tienen de los varios algoritmos criptográficos que aseguran muchos de los aspectos de su funcionamiento.

La forma más común de adquirir bitcoin (BTC) y otras criptomonedas es a través de exchanges (casas de intercambio de criptomonedas), mediante transferencias bancarias o tarjetas de crédito, a cambio de pequeñas comisiones. Cabe señalar que, mientras es posible enviar y recibir criptomonedas de una manera relativamente anónima, el intercambiarlas por dinero fiat (dinero por decreto) normalmente requiere revelar la identidad del usuario, la cual está ligada a una cuenta bancaria tradicional. Por este motivo, los exchanges generalmente están regulados por las leyes de los países en donde operan, para así poder evitar delitos de fraude o de lavado de dinero.

Otra opción para la adquisición de criptomonedas es a través de cajeros automáticos

especializados, siendo los de bitcoin los de mayor distribución, aunque aún poco difundidos en nuestro país, como se puede apreciar en la siguiente figura.

Figura 3-1

Cajeros bitcoin en México



Adaptado de *Bitcoin ATMs in Mexico* [Web], por Coin ATM Radar, 2022, Coin ATM Radar (<https://coinatmradar.com/country/138/bitcoin-atm-mexico/>). CC BY 2.0

Por otro lado, dependiendo de la implementación Blockchain utilizada, las criptomonedas pueden ser utilizadas para realizar transferencias directas entre personas, o bien pueden ser aprovechadas para desencadenar el funcionamiento de

procesos automatizados dentro de la implementación Blockchain, los cuales son mejor conocidos como smart contracts. Este último tópico se abordará a mayor profundidad en el siguiente capítulo.

3.3 ¿Qué es Bitcoin?

Bitcoin se puede definir de varias maneras; es un protocolo, una divisa digital y una plataforma. Es una combinación de red peer-to-peer formada por múltiples nodos distribuidos que se comunican entre sí mediante el protocolo Bitcoin y software que facilitan la creación y uso de la moneda digital, también llamada bitcoin. Hay que tener en cuenta que la palabra Bitcoin con la letra B mayúscula se usa para referirse al protocolo Bitcoin, mientras que bitcoin con la letra b minúscula se usa para referirse a bitcoin (BTC), la moneda digital (Bashir, 2017, p. 113).

Bitcoin es una criptomonedas descentralizada, criptográficamente segura y cuya tecnología subyacente es el Blockchain que, como se puede inferir de su nombre, su estructura de datos fundamental es la cadena de bloques.

En esta cadena, cada bloque contiene un identificador, la lista de transacciones que se llevaron a cabo y una referencia hacia el identificador del bloque anterior.

El bloque previo en esta cadena tiene también un identificador, una lista de transacciones y la referencia al identificador del bloque anterior. Esta relación continua sucesivamente hasta llegar al primer bloque de la cadena, al cual se le conoce como bloque génesis. Este bloque posee características especiales, ya que es el único bloque de la cadena que no contiene la referencia al identificador del bloque previo.

Bitcoin es una plataforma autónoma controlada por Software, no por personas, compañías o gobiernos. Si algún participante tuviera malas intenciones en ella, el software, el código y los algoritmos tienen reglas establecidas que no permitirían que

se comprometiera la red.

Tampoco se necesitan permisos especiales para su acceso; cualquiera puede ser parte de la red y puede proporcionarle más recursos computacionales o bien simplemente utilizarla como medio para transferir valor.

Bitcoin mantiene un registro público de activos y transacciones, donde cualquier persona puede ver que cuenta posee dichos activos y que transacciones se llevaron a cabo en el pasado, para así poder calcular su balance actual en bitcoin.

3.4 Wallets

Para poder adquirir bitcoin u otras criptomonedas se necesita una cartera (Wallet), la cual es un Software que permitirá su administración.

A alto nivel, un Wallet básicamente opera almacenando la llave privada del par de claves asimétricas asociadas a una criptomoneda y generadas para un usuario, lo cual le permitirá autenticar y firmar digitalmente las transacciones realizadas para dicho activo.

Dependiendo del tipo de Wallet, este incluirá formas para el envío y recepción de criptomonedas soportadas por la implementación, la consulta del balance actual para cada tipo de activo en posesión del usuario y el listado de las transacciones realizadas con anterioridad.

Al registrarse para obtener una cuenta dentro de un exchange, un usuario por lo general tendrá acceso a su Wallet dentro la misma plataforma en línea, en donde se podrán consultar las direcciones necesarias para la recepción de fondos para cierta criptomoneda en específico. Cada dirección corresponde a la llave pública del par criptográfico generado para cada activo, la cual es una cadena larga de caracteres

alfanuméricos. Esta misma dirección también se puede presentar como código QR.

Figura 3-2

Ejemplo de código QR y dirección bitcoin



Otro tipo común de Wallets son los basados en aplicaciones móviles. Estas son muy parecidas a las de los exchanges, sin embargo algunas se aprovechan de las bondades tecnológicas que brindan los teléfonos celulares, como la incorporación de la funcionalidad del lector óptico en la cámara para una rápida lectura de códigos QRs, lo que resulta en una interacción más fluida y una mejor experiencia de usuario al realizar una compra o una transferencia (un ejemplo práctico de ello se verá en un capítulo posterior).

En la figura 3-3, se puede observar un ejemplo de Wallet basada en aplicación móvil: la Chivo Wallet. Esta cuenta con más de un millón de usuarios ya que es la Wallet oficial del gobierno salvadoreño en los esfuerzos para la continuación de la estrategia de adopción de bitcoin, como moneda oficial en el país.

Figura 3-3

Chivo Wallet



Adaptado de *Chivo Wallet* [Web], por Gobierno de El Salvador, 2022, Chivo Wallet (<https://chivowallet.com/index.html>). Todos los derechos reservados 2022 por Gobierno de El Salvador.

Debido a que un Wallet puede ser algo tan simple como una llave privada, también es posible crear un Wallet de papel en sitios como <https://www.bitaddress.org/>,

donde desde el navegador del cliente se generará una dirección bitcoin junto con su clave privada asociada, las cuales se podrán imprimir y guardar en un lugar seguro.

Figura 3-4

Generación de Wallet de papel

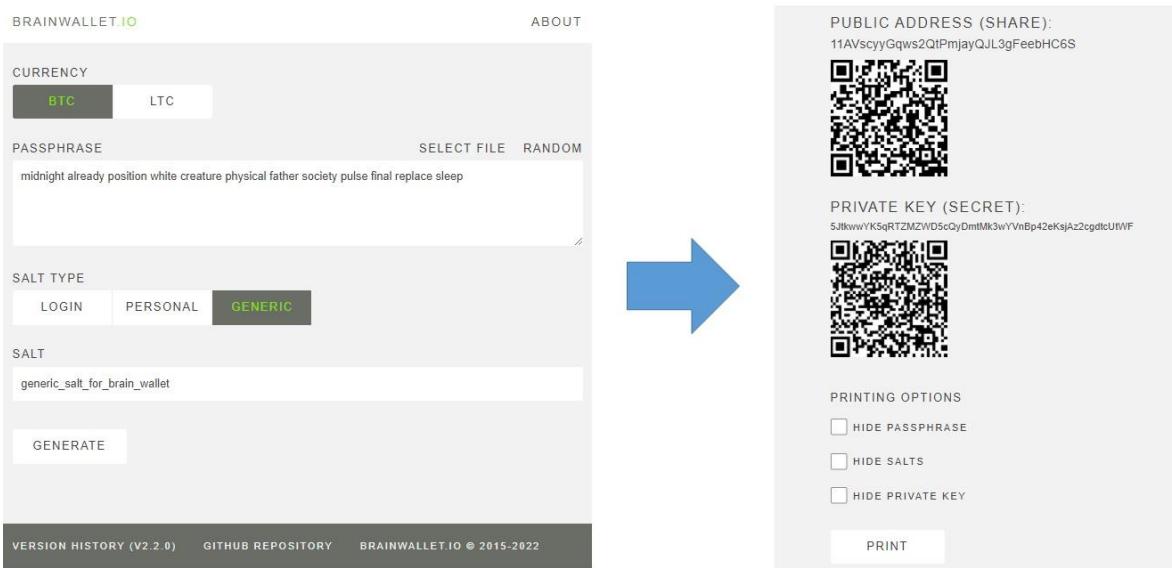


También existen las Wallets basadas en hardware, las cuales son pequeños dispositivos que almacenan las llaves privadas y las aislan de cualquier contacto con Internet, firmando las transacciones internamente.

Finalmente, también se puede generar una clave privada y su dirección pública asociada a través de sitios como <https://brainwallet.io/>, donde se utilizan una serie de palabras memorizadas por el usuario para calcular sus hashes y poder generarlas; a esta opción se le conoce como Brain Wallet.

Figura 3-5

Generación de Brain Wallet



Es muy importante señalar que para cualquier caso, se debe de tener especial cuidado con la clave privada de una dirección bitcoin, ya que si otra persona tiene acceso a ella, le sería posible autorizar las transacciones para gastar o transferir todos los fondos asociados a la misma, o en caso de que esta llegara a perderse, los fondos asociados se perderían definitivamente, ya que no existe manera alguna de recuperarla.

3.5 Minado de criptomonedas

Otro método utilizado para obtener criptomonedas es el minado. El minado es la forma en la que nuevas criptomonedas son creadas y generalmente asignadas proporcionalmente como recompensa para los nodos verificadores o mineros, a cambio del trabajo empleado para la validación de transacciones en el Blockchain (Proof of Work).

Por su parte, la red Bitcoin fue planeada para implementar un mecanismo conocido como halving. Este consiste en reducir la tasa de creación de nuevos bitcoin a la mitad cada cuatro años hasta que eventualmente se termine su distribución, lo cual según cálculos sucederá en el año 2140.

Hacia ese punto, los nodos mineros de bitcoin seguirían verificando las transacciones, sin embargo, solo podrían obtener la compensación asociada a las tarifas de las transacciones.

Finalmente, cabe señalar que en la actualidad la minería de criptomonedas que utilizan un mecanismo de Proof of Work como bitcoin ya no es rentable para un individuo común.

Figura 3-6

Hardware especializado para minado de criptomonedas



Esto se debe a los altos costos que implicaría adquirir un equipo con suficiente poder computacional y por consecuencia de un alto consumo energético, para poder competir contra grandes granjas de miles de equipos dedicados que cuentan con el Hardware más especializado para la realización de esta actividad (Figura 3-6).

3.6 Transacciones

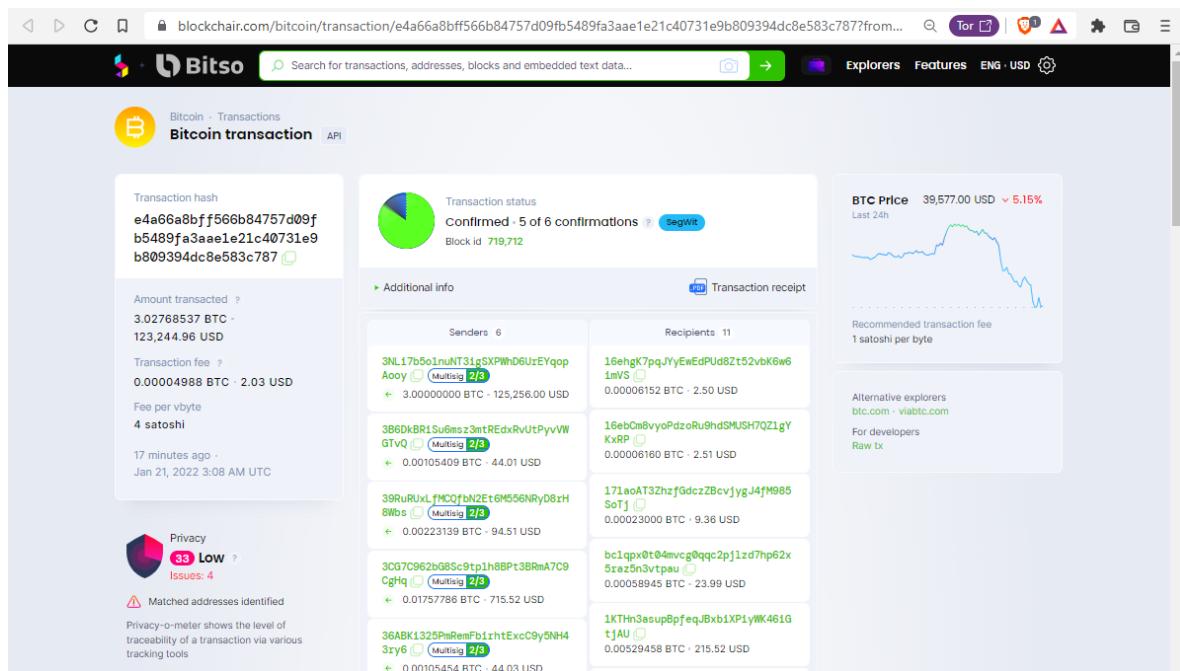
Una transacción es la unidad fundamental de una cadena de bloques. Una transacción representa una transferencia de valor de una dirección a otra (Bashir, 2017, p. 19).

Todas las transacciones son atómicas, lo que significa que todas las operaciones que la componen deben de completarse satisfactoriamente o en caso contrario, la transacción fallará.

Las transacciones se ejecutan independientemente y no interfieren una sobre otra, además, estas no pueden ser revertidas una vez que fueron registradas en el Blockchain, por lo que toda la información almacenada de un objeto sería también permanente.

Adicionalmente, las transacciones en el Blockchain no son cifradas, son visibles públicamente por cualquier persona y pueden llegar a ser totalmente anónimas, aunque esto solo podrá ser posible si se toman las medidas necesarias para no relacionar una clave pública (dirección Blockchain) hacia una identidad.

Por otra parte, hay que recordar que los bloques están formados de transacciones y estas pueden ser consultadas desde cualquier explorador de Blockchain en línea, como en el ejemplo de la figura 3-7.

Figura 3-7*Explorador de Blockchains: Blockchair.com*

Finalmente, según Bashir (2017), el ciclo de vida de una transacción en bitcoin consta de los siguientes pasos:

1. El remitente envía una transacción usando un Wallet.
2. El Wallet firma la transacción utilizando la clave privada del remitente.
3. La transacción se transmite a la red de Bitcoin mediante un algoritmo de inundación.
4. Los nodos mineros incluyen esta transacción en el siguiente bloque a minar.
5. El minado comienza una vez que el minero que resuelve el problema de Proof of Work transmite el bloque recién creado a la red.
6. Los nodos verifican el bloque y lo propagan más, y la confirmación comienza a generarse.
7. Finalmente, las confirmaciones comienzan a aparecer en el Wallet del

destinatario y aproximadamente después de seis confirmaciones, la transacción se considera finalizada y confirmada. La transacción puede considerarse definitiva incluso después de la primera confirmación, pero la idea de esperar seis confirmaciones es que se elimina virtualmente la probabilidad de un problema de double spending: utilizar los mismos activos en dos transacciones independientes mientras son verificadas al mismo tiempo (pp. 118-119).

4 Contratos inteligentes

4.1 ¿Qué son los contratos inteligentes o smart contracts?

La primera premisa que hay que tomar en cuenta al referirnos a un contrato inteligente, mejor conocido como smart contract, es que en realidad este no es un contrato ni tampoco es inteligente, sino más bien estamos hablando de código embebido y automatizable que se ejecuta sobre una red Blockchain.

En otras palabras y de acuerdo con Gupta (2017, p. 17), un smart contract es un acuerdo o conjunto de reglas que rigen una transacción comercial; se almacena en el Blockchain y se ejecuta automáticamente como parte de una transacción.

Además, señala que los smart contracts pueden tener muchas cláusulas contractuales que podrían ser parcial o totalmente autoejecutables, y que su propósito es brindar una seguridad superior a la del derecho contractual tradicional, al mismo tiempo que se reducen los costos y los retrasos inherentes de los contratos tradicionales.

Con los smart contracts, es posible ejecutar lógica de negocio automáticamente a medida de que los datos se van escribiendo en el Blockchain, lo cual lo convierte en una plataforma computacional distribuida a gran escala (Haunts, 2018b).

4.2 Plataforma Ethereum

Con características muy similares a las de Bitcoin, Ethereum es otra de las implementaciones Blockchain más grandes y establecidas en la actualidad.

Surgió de la propuesta de Vitalik Buterin en su whitepaper publicado en noviembre

de 2013 y su red fue lanzada públicamente el 30 de julio de 2015.

Siendo una plataforma pública y de código abierto, Ethereum permite la creación de aplicaciones descentralizadas a través de la utilización de smart contracts, de los cuales su soporte es una de las principales características de este Blockchain.

En marzo de 2017 se formó la Enterprise Ethereum Alliance, una organización sin fines de lucro formada por startups, investigadores y compañías del Fortune 500, dedicada a la investigación e impulso de soluciones Ethereum a nivel empresarial bajo un estándar abierto.

Todas las transacciones en la red Ethereum, así como las recompensas que los nodos mineros obtienen por su participación, son pagadas con la criptomoneda utilizada en este Blockchain: el Ether (ETH).

Por otra parte, la versión actual de Ethereum, Bitcoin, y muchos otros Blockchain utilizan el modelo de Proof of Work (PoW), donde los mineros deben competir para resolver complejos problemas criptográficos para poder validar las transacciones en el Blockchain.

Este proceso requiere de un poder computacional extremo que se ve reflejado a su vez en un alto consumo energético, lo cual ha sido fuertemente criticado debido al impacto ambiental que implica este modelo de minería.

El modelo de Proof of Stake (PoS) parece resolver los problemas del modelo de Proof of Work reemplazando el concepto de minería por completo, utilizando un mecanismo donde las transacciones son verificadas en función de la cantidad de tokens Ether que posee el usuario.

En Ethereum se pretende adoptar este modelo en su versión 2.0 (ETH2) en Junio de 2022, lo cual permitirá a la plataforma ser más escalable, segura y sostenible, reduciendo el consumo de energía hasta en un 99,95 %, en tanto que la minería de

Ethereum ya no generará ingresos (Kolakowski, 2021).

4.3 Aplicaciones descentralizadas

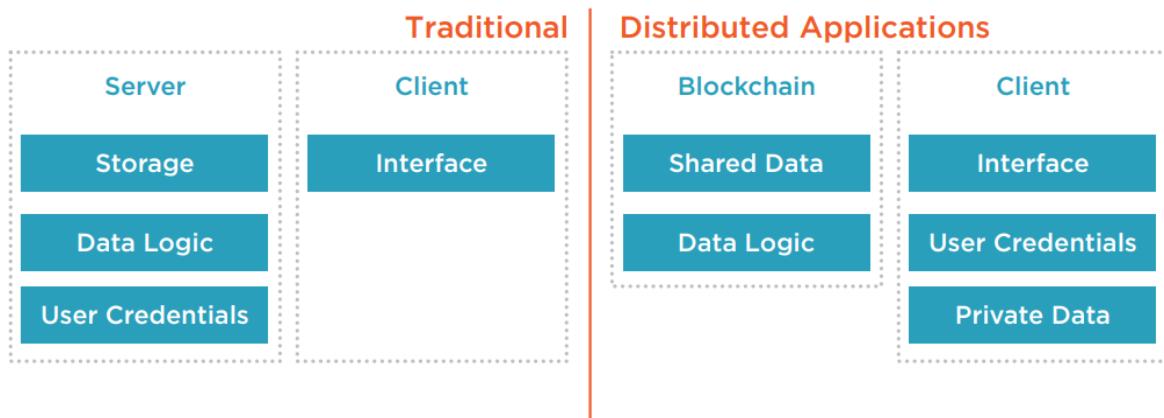
Una aplicación descentralizada, aplicación distribuida o DAPP, es una aplicación distribuida en un Blockchain y que puede ser ejecutada con los recursos computacionales de diferentes nodos, los cuales son propiedad de múltiples personas y organizaciones.

Cualquier participante puede proporcionar estos recursos a la red, sin embargo ninguno podrá tomar su control completo o parcial, por lo que se podrán construir aplicaciones que le saquen provecho a estas características.

En las aplicaciones descentralizadas, los datos compartidos están disponibles en el Blockchain y por ende existen en muchas ubicaciones, por lo que no importa si uno de los nodos falla. Lo mismo se aplica a la lógica de negocio, donde la lógica se comparte en el Blockchain. Así, tan pronto como se cargue un contrato en el Blockchain, este se propagará y se ejecutará de la misma manera, independientemente de dónde se haya ejecutado. El cliente en una configuración descentralizada también opera de manera diferente, ya que este será el responsable de almacenar sus propias credenciales de usuario y por lo general, también almacenará más datos de la aplicación (Sandberg, 2018).

Figura 4-1

Arquitectura de aplicaciones tradicionales y descentralizadas



Adaptado de *Blockchain Fundamentals* [Curso en línea], por J. Sandberg, 2018, Pluralsight (<https://app.pluralsight.com/library/courses/blockchain-fundamentals>). Todos los derechos reservados 2018 por Pluralsight.

Entre algunas de las ventajas que se pueden encontrar al construir aplicaciones descentralizadas tenemos que:

- Son más confiables, ya que no tienen un punto único de fallo y aunque algunos nodos no estén habilitados, la aplicación continuará ejecutándose.
- Son más seguras, ya que el código y la información están protegidos criptográficamente en el Blockchain, por tanto nadie podrá alterarlos.
- Son transparentes, ya que cada participante podrá verificar que las reglas se cumplan y que nadie intente modificar información que no debería ser cambiada.

4.4 Modelo de pago: Gas

El modelo de pago en la plataforma Ethereum se basa en un concepto conocido como Gas, el cual corresponde a la tarifa que hay que pagar por ejecutar contratos o transacciones en esta red.

La cantidad de Gas es proporcional a los recursos computacionales, de memoria o de almacenamiento requeridos durante la ejecución de una transacción y fue diseñada de esta manera para evitar abusos en la red, como inundación de transacciones por spam, ejecución de bucles infinitos, etc.

El Gas se mide en fracciones de Ether, siendo el wei (representando 10^{-18} Ether) la unidad más pequeña y el gwei (representando 10^{-9} Ether) la unidad más comúnmente utilizada.

El Gas Price representa la tarifa que el remitente desea pagar a los mineros por unidad de Gas utilizada; a mayor disposición de pagar, mayor prioridad le asignarán los mineros a la transacción y por consecuencia, será más rápida.

Por otro lado, el Gas Limit representa la máxima cantidad de unidades de Gas que el remitente está dispuesto a utilizar durante toda la transacción. Este monto también es importante para los mineros, ya que los ayuda a calcular una estimación de la ganancia al realizar la transacción.

Al momento de iniciar una transacción, el remitente tendrá que establecer suficiente Gas o de lo contrario la ejecución se interrumpirá, se revertirán los cambios realizados y el Gas utilizado será consumido, lo que representaría un desperdicio de dinero para el remitente.

Así mismo, el remitente comprará la cantidad total de Gas (Gas Limit) por adelantado al inicio de la transacción, y al final de esta se le reembolsará la cantidad

de Gas que no haya sido utilizada.

Finalmente, hay que tomar en cuenta que la diferencia entre el Gas Limit y el Gas realmente consumido no debe ser muy grande, ya que esto podría estar sujeto a penalizaciones por parte de los mineros, al esperar obtener una recompensa mayor.

Para lidiar con estas condiciones existen Wallets como MetaMask, la cual es una extensión de navegador que nos permite la interacción con el Blockchain de Ethereum y que realiza el cálculo de la cantidad de Gas apropiada para una transacción específica antes de su ejecución, o bien, se pueden utilizar calculadoras de Gas en línea como el sitio <https://ethgasstation.info/>.

4.5 Transacciones en Ethereum

Una transacción en Ethereum está formada por el receipt, la firma que identifica al remitente, el valor a ser transferido, el valor de Gas Price, el valor de Gas Limit y un campo de datos opcional que puede contener el mensaje enviado a un contrato.

Las transacciones en Ethereum solo pueden ser iniciadas desde cuentas, las cuales pueden ser de dos tipos:

- Externally Owned Account (EOA), que básicamente está asociada a un usuario que quiere realizar alguna transacción, y
- Contract Account, que está asociada con una entidad de smart contract y posee código, el cual se desencadena cuando un smart contract recibe transacciones o mensajes desde otros smart contracts. Este código puede contener instrucciones para enviar Ether, llamar funciones de otros contratos, leer o escribir desde su propio contenido, consultar información de la transacción actual, entre otros.

4.6 Desarrollo en Ethereum

4.6.1 Ethereum Virtual Machine

Todos los nodos participantes en la red deben correr la Ethereum Virtual Machine (EVM), la cual es el ambiente de ejecución de Ethereum que sabe cómo verificar y ejecutar los smart contracts que fueron construidos sobre la plataforma de Blockchain.

La EVM también es responsable de calcular que tan compleja puede llegar a ser una transacción y cuál debería ser el precio a pagar por ejecutarla. A esta acción se le conoce como cálculo de Gas consumption.

Adicionalmente, la EVM también verificará las transacciones antes de que lleguen a ser parte del registro permanente de Blockchain.

4.6.2 Lenguaje de programación Solidity

Para crear un smart contract y que este sea posteriormente publicado en la red Ethereum, primero se debe de crear un programa computacional en un lenguaje especializado y soportado por la EVM, como lo es Solidity.

Solidity es un lenguaje de programación de alto nivel, orientado a objetos, con una sintaxis muy parecida a la del lenguaje JavaScript y específicamente creado para desarrollar smart contracts.

El código fuente en lenguaje Solidity para este smart contract, se compila localmente y se transforma en Ethereum bytecode.

4.6.3 Redes en Ethereum

Posteriormente, este se publica hacia una de las redes Ethereum públicas disponibles, ya sea para un ambiente real (involucrando transacciones de valor reales con Ether) como la red Mainnet, un ambiente Testnet o de pruebas como las redes Rinkeby y Ropsten (donde se usa Ether de prueba) o bien para un entorno completamente local.

Es importante señalar que una misma cuenta en Ethereum podrá funcionar para los diferentes ambientes, sin embargo el balance de Ether y el historial de transacciones son independientes, por lo que no se podrán transferir entre redes.

Una vez que el smart contract fue publicado, se podrá desencadenar su ejecución mediante la realización de una transacción a través del Wallet de preferencia.

Finalmente, el smart contract será ejecutado en la EVM del nodo que valide dicha transacción. Este último paso se repetirá tantas veces como sea utilizado el contrato.

5 Propuesta de implementación

5.1 Objetivos de la propuesta

Los objetivos de esta propuesta son la elaboración de una guía funcional para la implementación de un mecanismo de pago con criptomonedas y el desarrollo de una prueba de concepto de un contrato inteligente.

5.2 Alcances y limitaciones

Debido a que esta propuesta está orientada hacia emprendedores y MiPyMes en México, esta se enfocará a los servicios ofrecidos por el exchange mexicano: Bitso, no obstante los pasos para su implementación serán muy similares aún si se utiliza un exchange diferente o bien si se realizan desde otro país.

Adicionalmente, la implementación enfocará los ejemplos mostrados al pago con bitcoin por ser la criptomoneda más ampliamente utilizada, sin embargo los pasos para realizar pagos con otras criptomonedas soportadas por el exchange serán muy parecidos.

Por otro lado, existen muchas alternativas al elegir una plataforma y sus componentes adecuados para el desarrollo de un contrato inteligente, sin embargo esta propuesta solamente se enfocará en el desarrollo específico de una prueba de concepto que sirva de plantilla para la innovación de nuevos casos de uso, utilizando para ello la plataforma Ethereum, la red de prueba Ropsten, el lenguaje de programación Solidity y el Wallet MetaMask.

5.3 Adopción de mecanismo de pago con criptomonedas

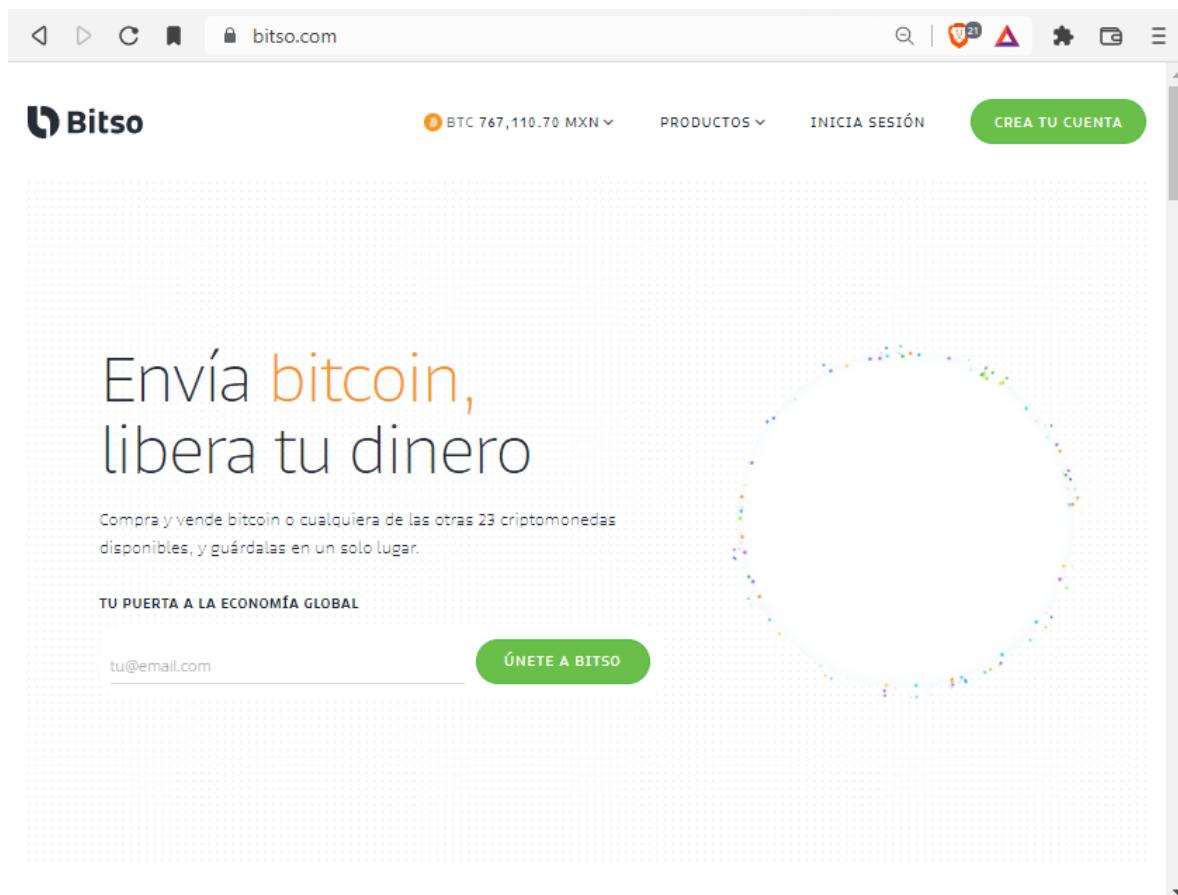
5.3.1 Creación de cuenta de usuario en el exchange Bitso

Para crear una cuenta de usuario en el exchange Bitso, en donde se tendrá acceso a una Wallet para poder realizar transacciones de compra y venta de criptomonedas, el emprendedor o microempresario tendrá que seguir los siguientes pasos:

1. Navegar hacia la página principal del exchange en <https://bitso.com/> y seleccionar la opción para crear cuenta.

Figura 5-1

Bitso: página principal



2. A continuación aparecerá un formulario de registro que solicitará un correo electrónico como nombre de usuario y una contraseña, además de poder seleccionar el tipo de cuenta a crear, ya sea personal o empresarial.

Figura 5-2*Bitso: formulario para crear una cuenta*

The screenshot shows the Bitso account creation page. The title is "Crea tu cuenta". There are two tabs: "Personal" (selected) and "Empresarial". Below the tabs, there's a message: "Estás a unos pasos de entrar al mundo cripto." The form fields include:

- País de residencia: México
- Correo electrónico: [REDACTED]
- Regístrate con el correo que más utilizas
- Crea una contraseña: [REDACTED]
- Min. 8 caracteres con números y símbolos
- Confirma tu contraseña: [REDACTED]

Checkboxes at the bottom:

- Acepto la Política de privacidad y los Términos y condiciones de Bitso y tengo al menos 18 años
- Acepto el Aviso de privacidad y Términos y condiciones de Nvio Pagos México SAPI de CV Institución de Fondos de Pago Electrónico
- Quiero conocer lo más reciente de Bitso

At the bottom, there's a "I'm not a robot" reCAPTCHA field with the text "reCAPTCHA Privacy - Terms".

Cabe señalar que las diferencias entre estos tipos de cuenta al momento de escribir esta investigación son algunos beneficios adicionales para las cuentas empresariales, como límites de operación más altos y recepción y envío de pagos hacia terceros. Esto implicará presentar documentación oficial de la

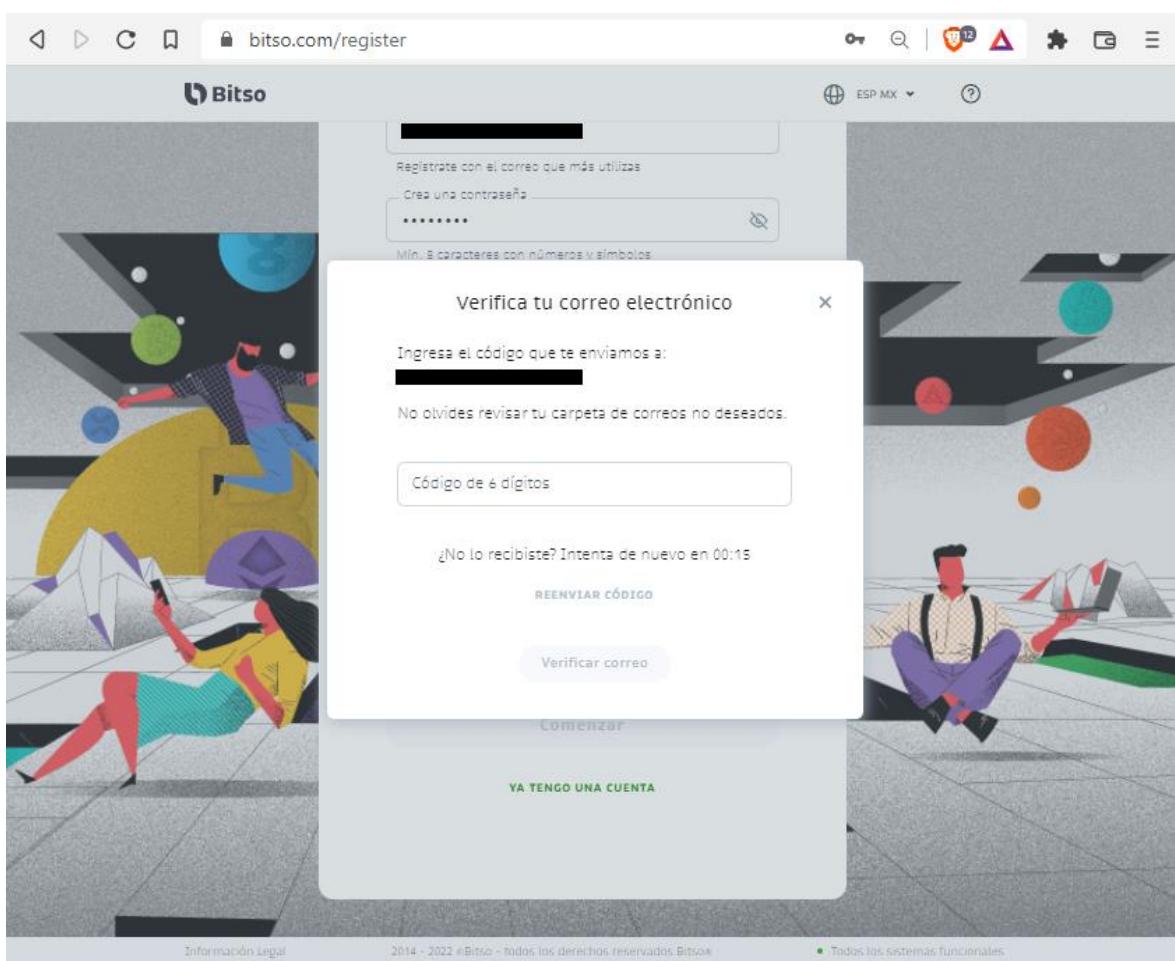
empresa como su acta constitutiva, RFC, comprobantes, constancias entre otros.

Para efectos de esta investigación se procederá a crear una cuenta personal, sin embargo, para conocer a detalle los requisitos, lineamientos regulatorios, beneficios y los diferentes niveles para límites operativos de una cuenta empresarial, se podrán consultar los enlaces correspondientes incluidos en la bibliografía.

3. Posteriormente, tras aceptar las políticas y avisos de privacidad, los términos y condiciones y el CAPTCHA, el sitio solicitará un código de verificación enviado vía correo electrónico para confirmar la dirección registrada.

Figura 5-3

Bitso: verificación de correo electrónico

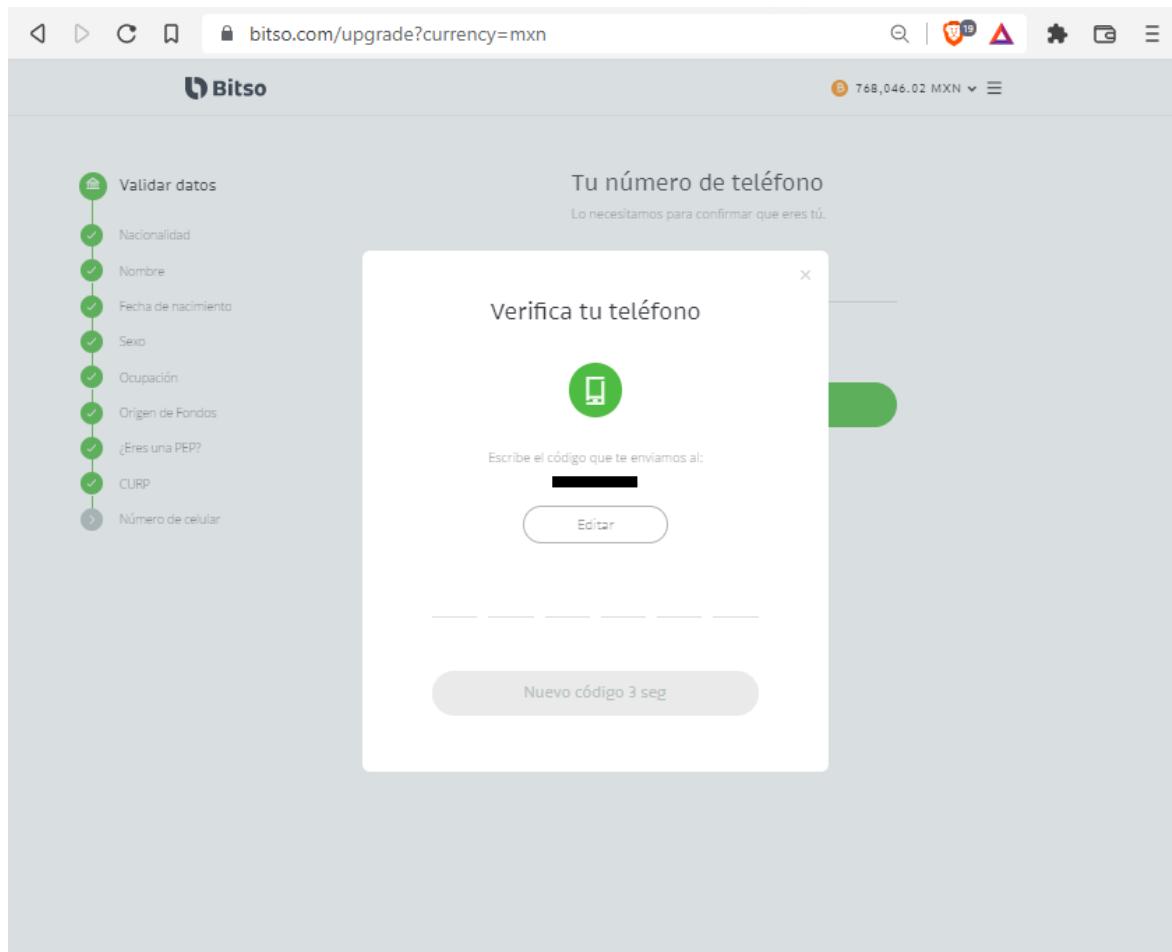


4. Despu s de esta verificaci n, el sitio solicitar  introducir informaci n adicional del usuario como fecha de nacimiento, CURP, n mero telef nico, etc. Esta informaci n deber  ser v lida para poder ligarse a la creaci n de una cuenta del Sistema de Transferencias y Pagos (STP), que servir  como medio para ingresar fondos v a SPEI.

Figura 5-4*Bitso: validaci n de datos*

The screenshot shows a web browser window for the Bitso website at bitso.com/upgrade?currency=mxn. The title bar includes the Bitso logo and a balance of 767,478.01 MXN. The main content is titled 'Validar datos' (Validate data) and lists several fields: Nacionalidad, Nombre, Fecha de nacimiento, Sexo, Ocupaci n, Origen de Fondos, ¿Eres una PEP?, CURP, and N mero de celular. The 'Nacionalidad' section is expanded, showing dropdown menus for '¿Cu l es tu pa s de nacionalidad?' (Mexico) and '¿En qu  pa s naciste?' (Mexico), along with a 'Estado de Nacimiento' dropdown menu. A large 'Siguiente' (Next) button is visible at the bottom.

5. A continuaci n, se solicitar  un c digo de verificaci n enviado v a SMS para confirmar el n mero telef nico registrado.

Figura 5-5*Bitso: verificación de número telefónico*

6. Para este punto, ya se contará con una cuenta de nivel 1, con límites de fondeo y retiro de \$4,500.00 MXN y saldo máximo de hasta \$6,000.00 MXN. Para incrementar el nivel de la cuenta, más requisitos de documentación serán solicitados, los cuales podrán ser consultados en el enlace correspondiente incluido en la bibliografía.

Figura 5-6*Bitso: sección de límites de cuenta*

The screenshot shows the Bitso user interface. The left sidebar has a green highlight on the 'LÍMITES' (Limits) option. The main content area is titled 'Límites de Cuenta' (Account Limits). It shows the 'Crypto' tab is selected. Under 'Bitcoin', it says 'Depósitos mensuales Ilimitado' (Monthly deposits unlimited) and 'Retiros mensuales 0.00000000 de 0.00617129 BTC' (Monthly withdrawals 0.00000000 to 0.00617129 BTC). A modal window titled 'Incrementa los límites de tu wallet' (Increase the limits of your wallet) is displayed, containing instructions and a list of required documents:

- Límite de depósito mensual hasta 0.00000000 BTC
- Límite de retiro mensual hasta 0.00000000 BTC
- Para aumentar tus límites, tenemos que conocerte un poco más. Te haremos las siguientes preguntas:
- Dirección
- Identificación oficial ?
- Comprobante de domicilio ?

Aumentar límites

Information at the bottom: Información Legal, 2014 - 2022 eBitso - todos los derechos reservados Bitso®, Todos los sistemas funcionales.

- Finalmente, también se podrá acceder a la página principal del Wallet, en donde se reflejarán los balances en pesos mexicanos (MXN) y en criptomonedas que tengamos disponibles al momento.

Figura 5-7*Bitso: Wallet*

The screenshot shows the Bitso Wallet interface. On the left, a sidebar displays a combined balance of **0.00** MXN. Below this, there's a note: "El balance total en MXN es un aproximado al último precio de 03/02/2022 11:10 PM." A link "Ir a historial de movimientos" is also present. On the right, a "Monedas" section lists various cryptocurrencies with their current values in MXN:

Crypto	Value (MXN)
MXN	0.00
BTC	0.00000000
AAVE	0.00000000
AXS	0.00000000
Other	0.00000000

Below this, a specific section for "Pesos mexicanos MXN" shows a balance of **0.00 MXN**. At the bottom of the page, there are links for "Depositar", "Enviar", and "Convertir". The footer contains links for "COMPAÑIA", "RECURSOS", and "PRODUCTOS", along with a language selection dropdown set to "Español MX".

8. Adicionalmente, será altamente recomendable acceder a la sección de seguridad para habilitar y configurar los mecanismos de autenticación de dos factores (2FA) y NIP de transacción.

Figura 5-8*Bitso: seguridad de cuenta*

The screenshot shows the Bitso account security settings interface. On the left, a sidebar menu lists options: RESUMEN, SEGURODAD (selected), CONFIGURACIÓN, LÍMITES, NOTIFICACIONES, MONEDAS, AUTOVENTA, HISTORIAL, DISPOSITIVOS, API, and BENEFICIARIOS. The main content area is titled 'Seguridad de cuenta' with the sub-instruction 'Configura la seguridad de tu cuenta.' It shows that 'Token de dos factores (2FA)' is active, indicated by a green 'Token 2FA activado' button and a 'Desactivar' button. Below this, there's an email section with 'Correo electrónico' set to 'gerardo.alberto@gmail.com' and a 'Cambiar' button. Further down is a password section with a 'Contraseña' input field and a 'Cambiar' button. At the bottom, there's a transaction PIN section with 'NIP de transacción' and a 'Modificar' button. The top right corner shows a balance of 'B 859,349.18 MXN'. The bottom of the page includes standard footer links like 'Información Legal', '2014 - 2022 eBitso - todos los derechos reservados Bitso', and a status bar indicating 'Todos los sistemas funcionales'.

5.3.2 ¿Cómo recibir un pago con bitcoin?

Para recibir pagos con bitcoin, el usuario proveedor de algún bien o servicio tendrá que proporcionar su código QR o dirección bitcoin al cliente. Para ello, deberá elegir una de las siguientes alternativas, las cuales están disponibles en el exchange Bitso al momento de escribir esta investigación.

5.3.2.1 Dirección bitcoin de autoventa

Este método consiste en generar una dirección estática bitcoin en la que, una vez que se reciban montos de bitcoin, estos serán vendidos (al precio actual del mercado) y convertidos automáticamente a la moneda local previamente definida, después de las confirmaciones necesarias por el Blockchain.

De esta manera, el monto del pago previamente acordado no se depreciará debido a las altas fluctuaciones a las que la mayoría de las criptomonedas están sujetas.

Por otro lado, una desventaja de este método radica en que al tratarse de una dirección fija, cualquier persona podrá rastrear todas las transacciones asociadas a la misma dentro del Blockchain, por lo que se recomienda únicamente para escenarios muy específicos donde esto no represente mayor problema.

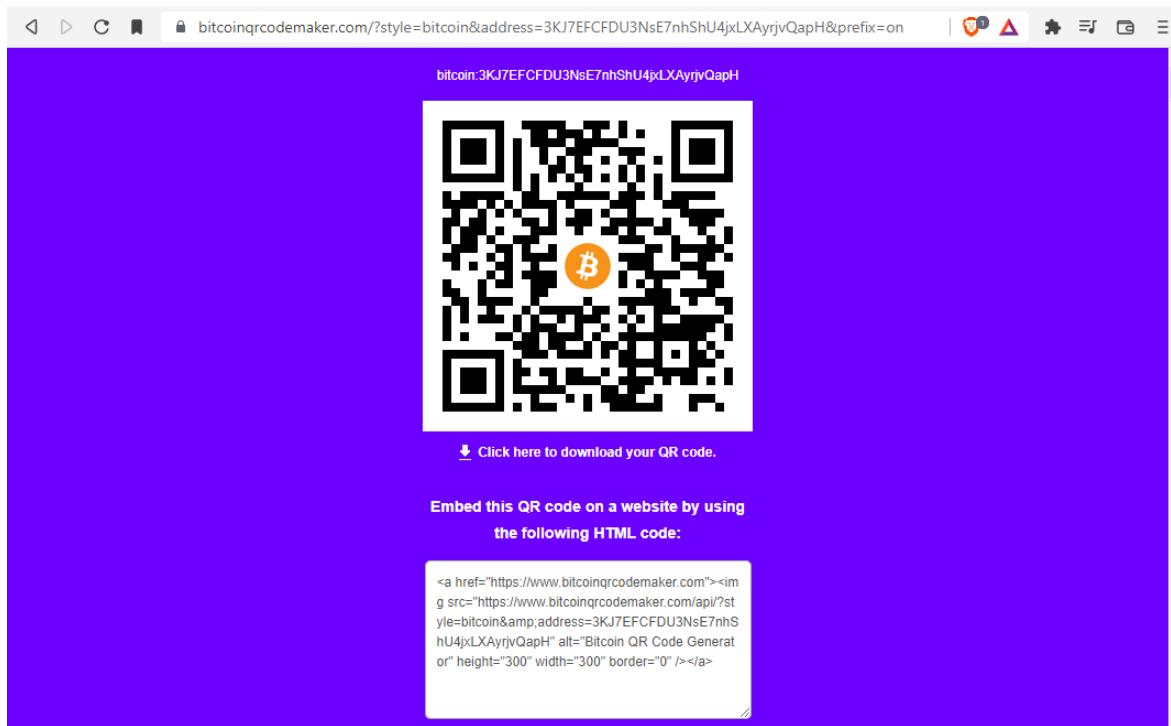
Para obtener la dirección bitcoin de autoventa, se sigue el siguiente procedimiento:

1. El usuario deberá acceder a la sección de venta automática de bitcoin, donde elegirá la moneda local de su conveniencia y hará click en el botón generar. Con esto, la dirección para venta automática será generada y desplegada en la parte inferior de esta sección.

Figura 5-9*Bitso: venta automática de bitcoin*

The screenshot shows the Bitso website interface. On the left, a sidebar menu lists various options: RESUMEN, SEGURIDAD, CONFIGURACIÓN, LÍMITES, NOTIFICACIONES, MONEDAS, AUTOVENTA (which is highlighted), HISTORIAL, DISPOSITIVOS, API, and BENEFICIARIOS. The main content area is titled "Venta automática de bitcoin" and explains how to use it to send Bitcoin from an external wallet to Bitso for immediate conversion to MXN. Below this, there's a section for generating a local currency address for automatic sales, with fields for "Moneda local" set to "MXN". A "Generar" (Generate) button is present. At the bottom, a section titled "Direcciones Activas" (Active Addresses) shows a single active address: "BTC para MXN" followed by a long alphanumeric string.

2. Opcionalmente, se copiará la dirección bitcoin generada en el paso anterior y se accederá al sitio <https://www.bitcoinqrcodemaker.com/>, donde de manera externa al exchange se podrá generar el código QR asociado a la misma.

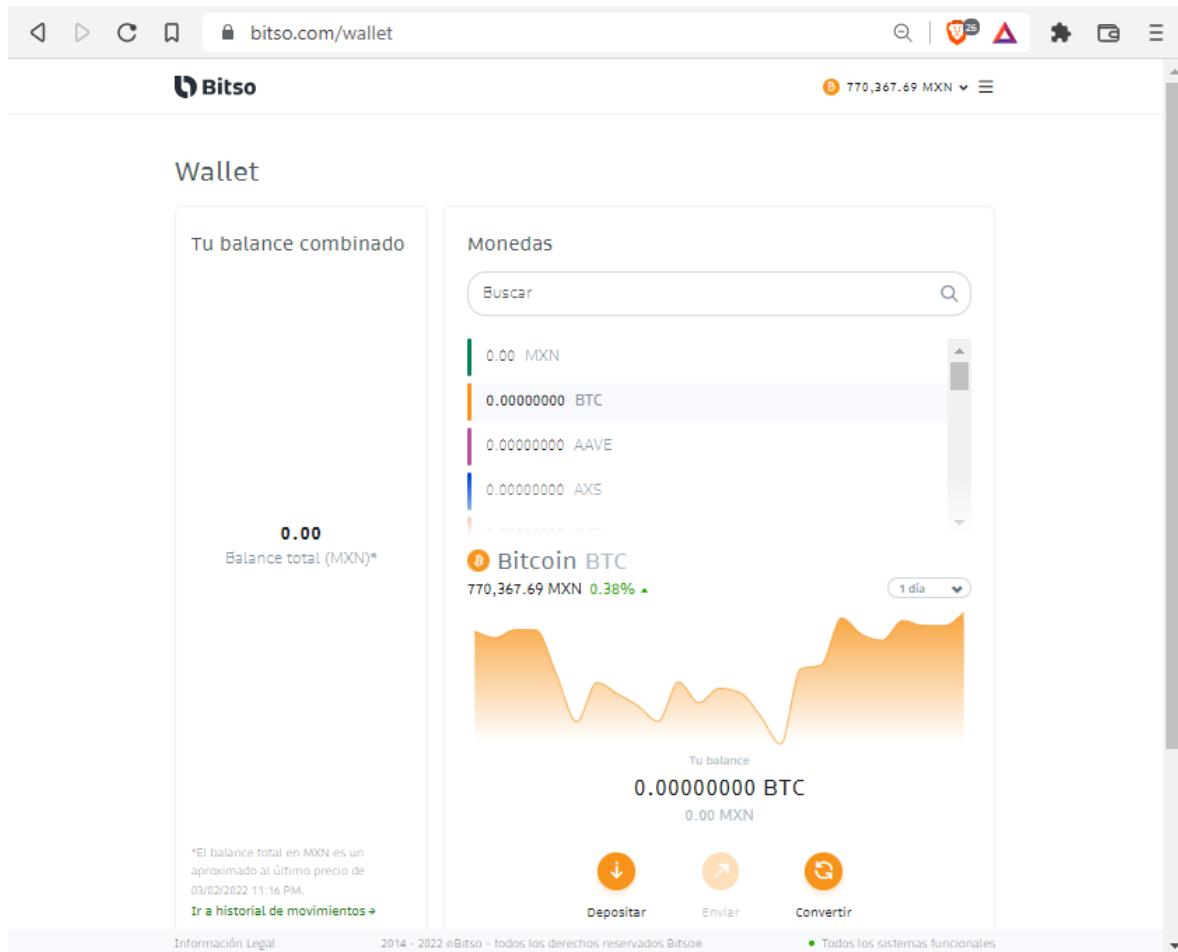
Figura 5-10*Sitio para generar códigos QR bitcoin*

5.3.2.2 Dirección bitcoin dinámica

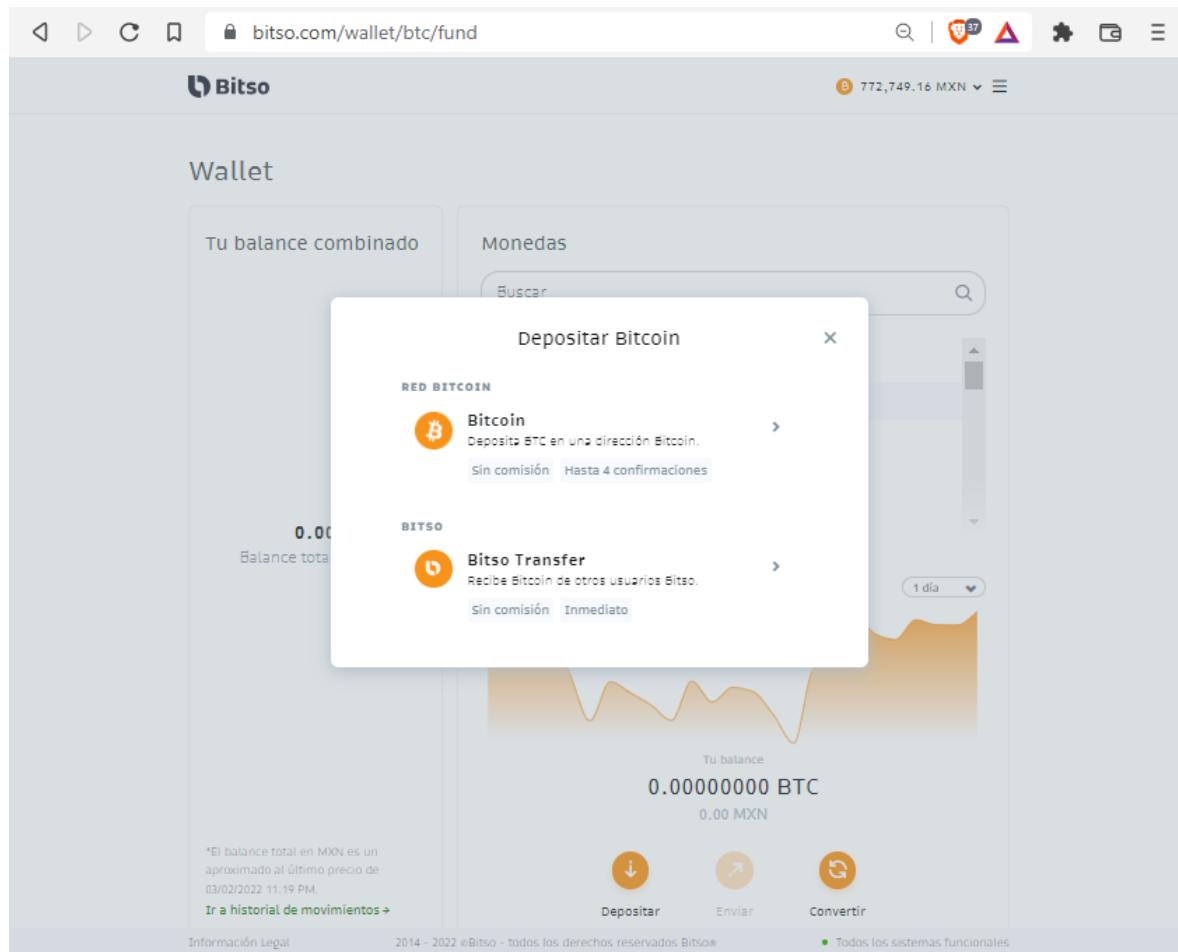
Este método consiste en obtener una dirección bitcoin que será cambiada por el exchange cada vez que se realice un nuevo depósito. Esto brindará mayor privacidad para el usuario debido a que las transacciones no estarán ligadas directamente a una única dirección, haciendo más complicado su rastreo.

Para obtener la dirección bitcoin dinámica, se sigue el siguiente procedimiento:

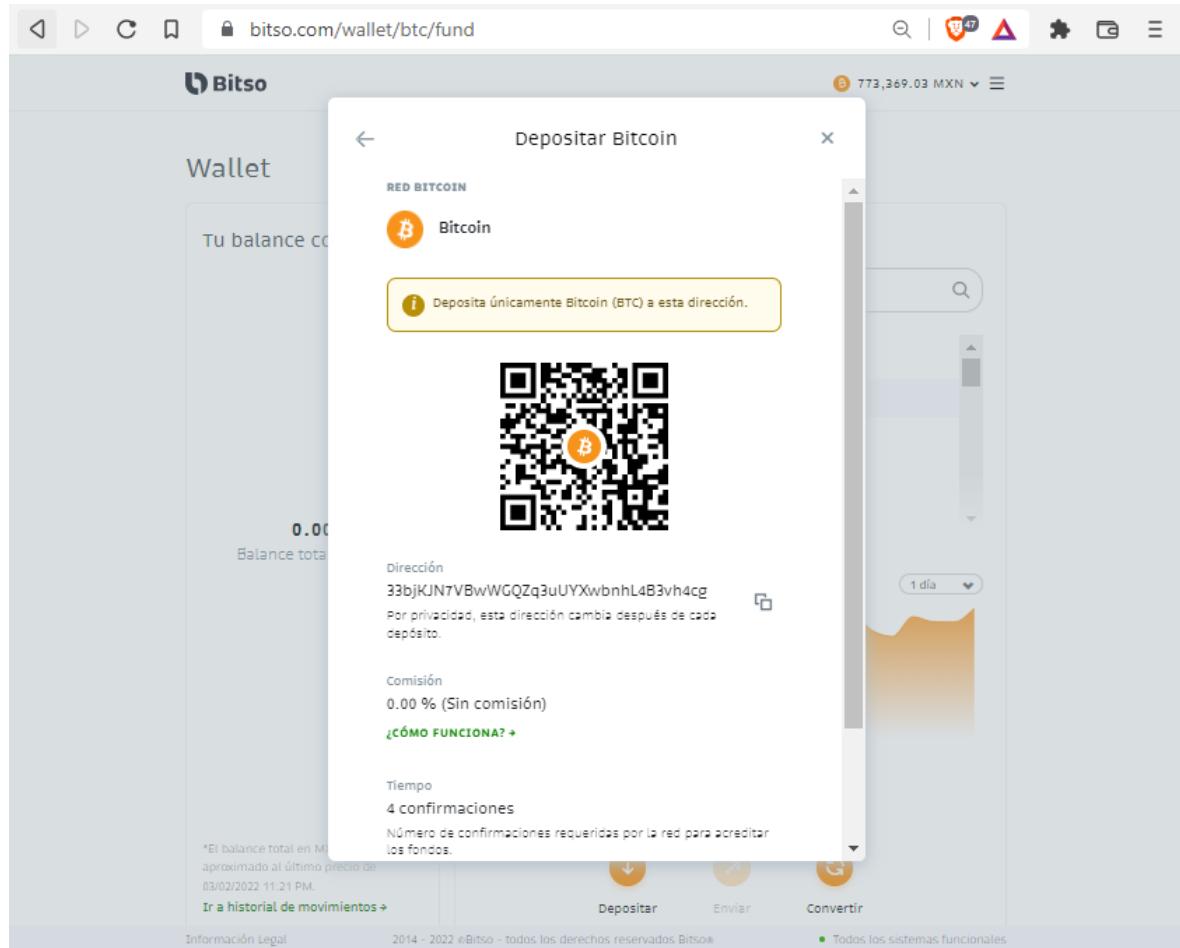
1. El usuario deberá acceder a la sección de Wallet, seleccionará bitcoin (BTC) de la lista de criptomonedas disponible y hará click en la opción depositar.

Figura 5-11*Bitso: Wallet de bitcoin (BTC)*

2. Acto seguido, el usuario deberá elegir la opción para depositar bitcoin hacia una dirección dentro de la red Bitcoin.

Figura 5-12*Bitso: opciones para depositar bitcoin*

3. Finalmente, la plataforma proporcionará la dirección bitcoin y su código QR asociado.

Figura 5-13*Bitso: dirección bitcoin (BTC) y código QR para depósito*

5.3.3 ¿Cómo realizar un pago con bitcoin?

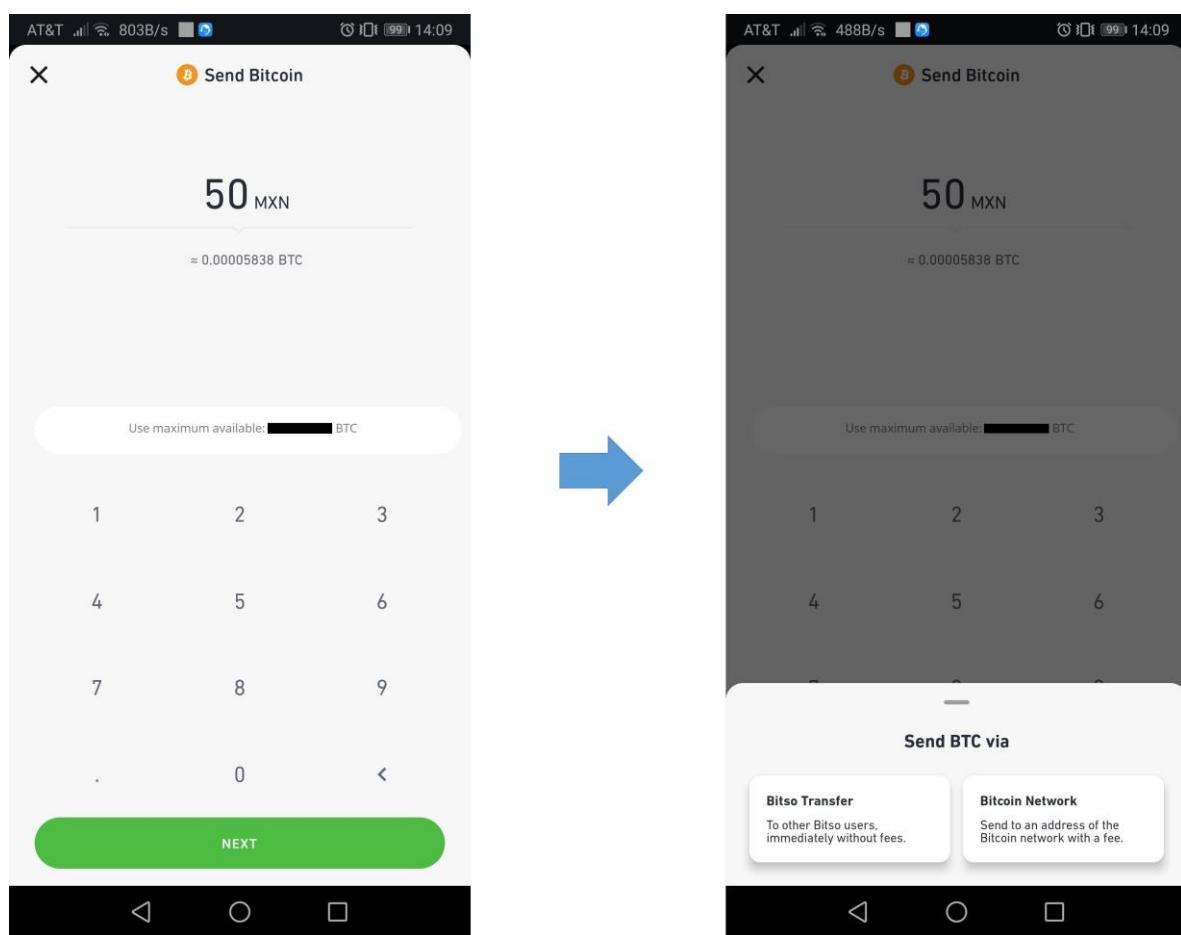
Para realizar una operación de pago con bitcoin, se siguen los siguientes pasos:

1. El usuario proveedor del bien o servicio, tendrá que proporcionar su código QR o dirección bitcoin al cliente. Para este ejemplo, utilizaremos su dirección bitcoin dinámica (ver Figura 5-13).

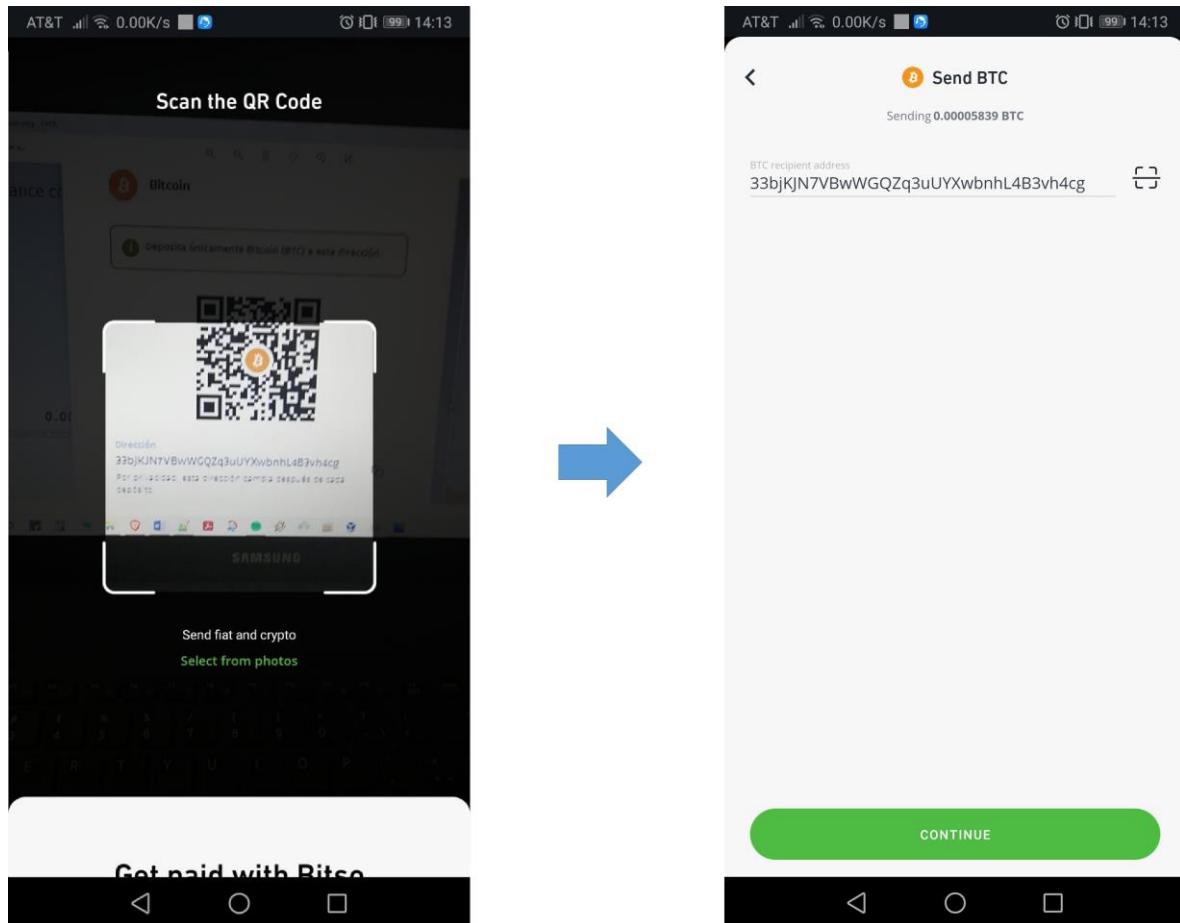
2. Por otro lado, el cliente deberá abrir su aplicación móvil de Wallet, elegirá la opción para envío de bitcoin y tendrá que introducir el monto pactado a ser transferido, ya sea eligiendo el monto en MXN o bien la cantidad en bitcoin. Para ambos casos se mostrará el equivalente entre esta paridad (MXN/BTC), utilizando el tipo de cambio al momento de haber iniciado la operación, ya que el valor de las criptomonedas puede ser altamente fluctuante.

Figura 5-14

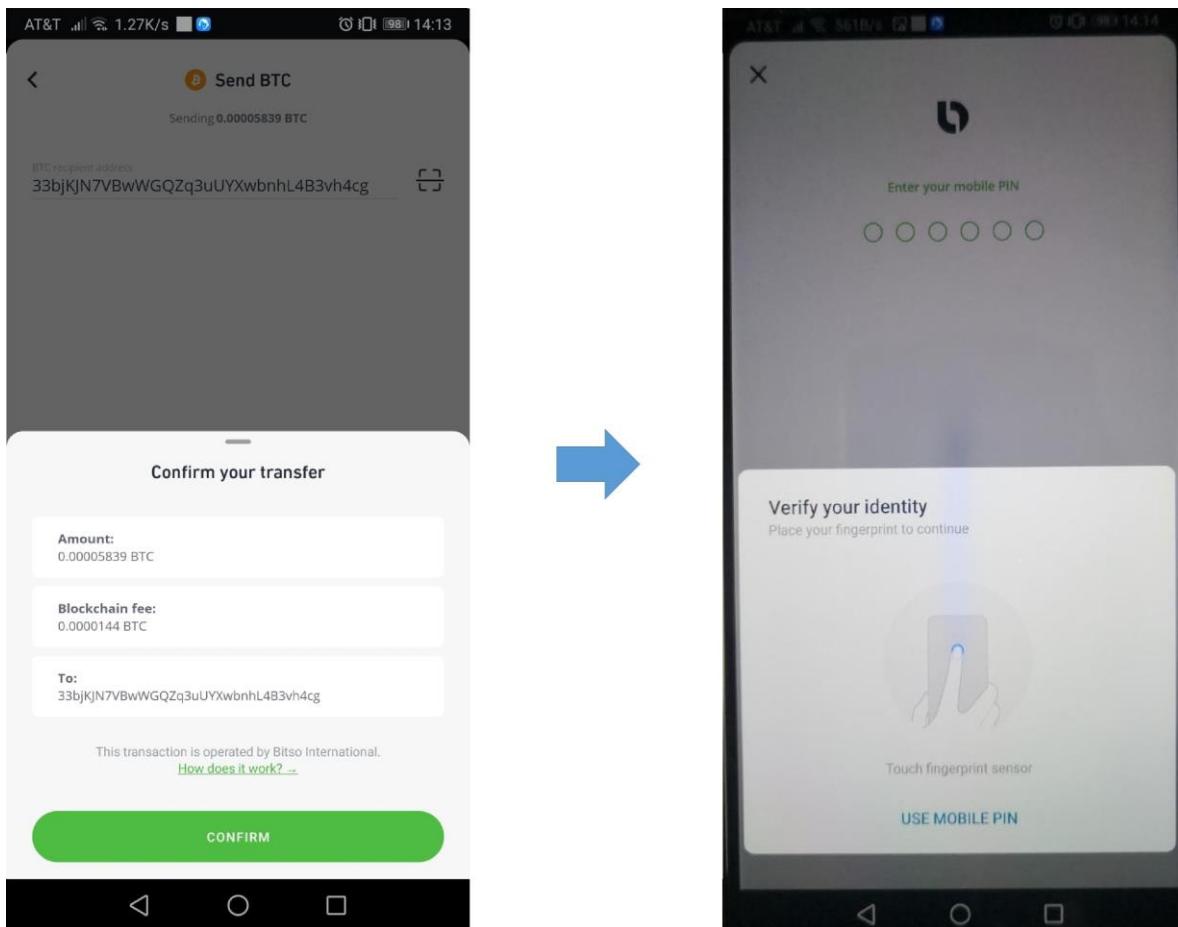
Envío de bitcoin: definición de monto y selección de red



3. Después de haberse asegurado de elegir la red Bitcoin, el cliente deberá escanear el código QR proporcionado a través de la funcionalidad de lector óptico de su aplicación móvil y verificará la dirección resultante.

Figura 5-15*Envío de bitcoin: lectura de código QR y verificación de dirección*

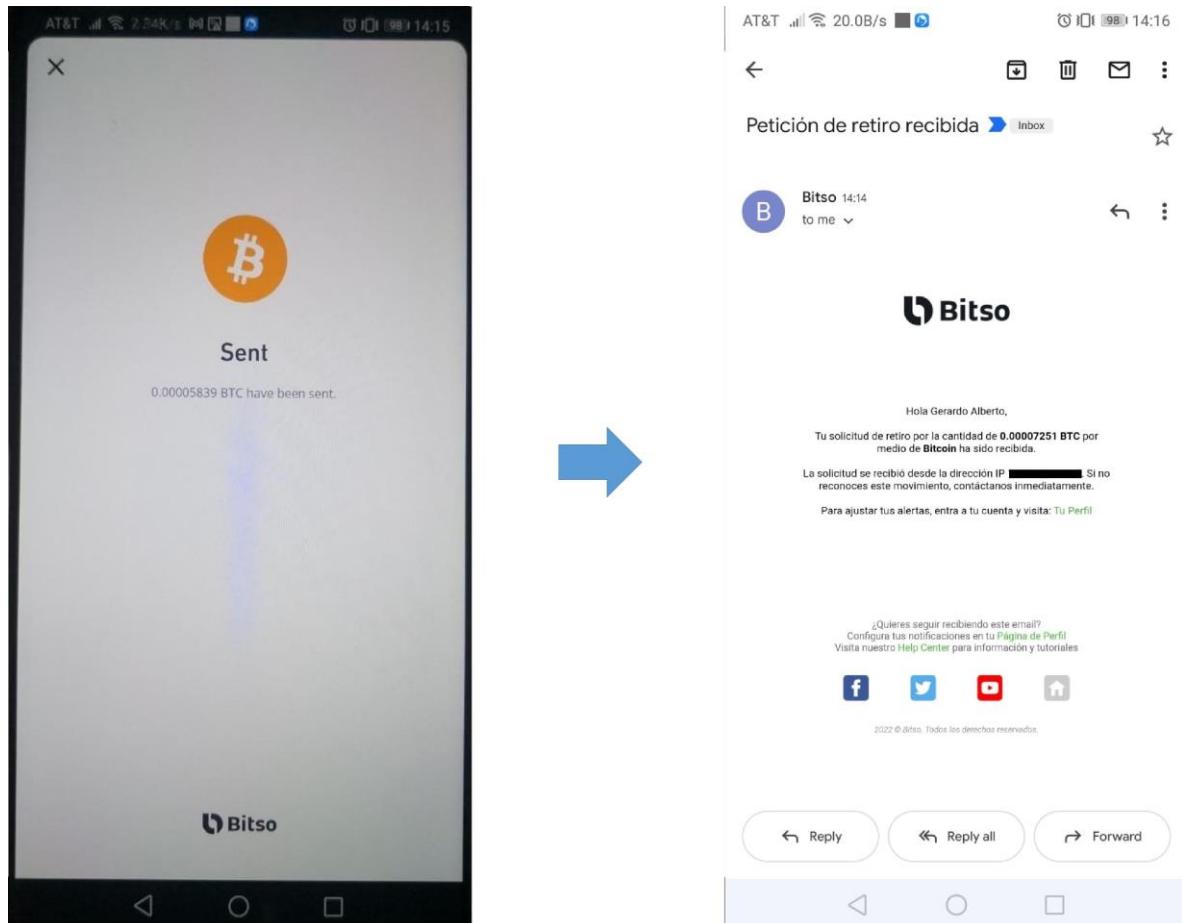
4. Una vez que el cliente haya verificado los montos de la transacción, incluyendo la comisión por la operación, el cliente procederá a confirmar la transferencia y posteriormente, a verificar su identidad para completar el proceso.

Figura 5-16*Envío de bitcoin: confirmación de transferencia y verificación de identidad*

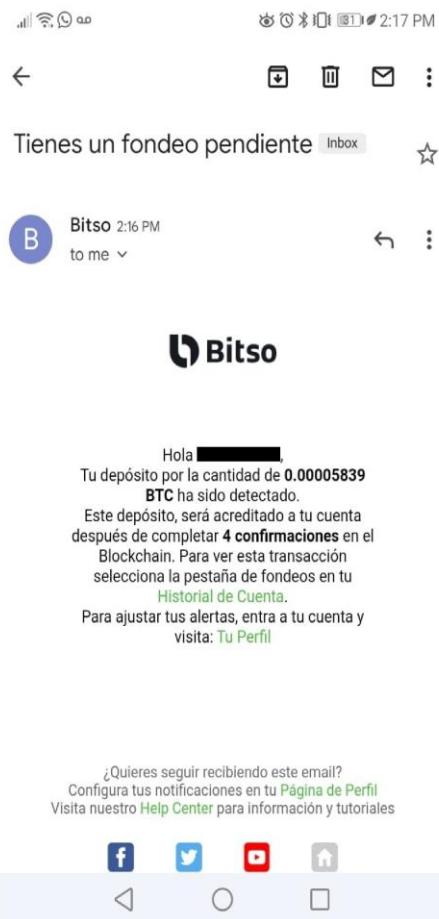
5. En este punto, el cliente habrá completado la operación de envío de bitcoin, por lo que recibirá una notificación push en su teléfono y/o un correo electrónico (dependiendo de su configuración de alertas) informando de la misma.

Figura 5-17

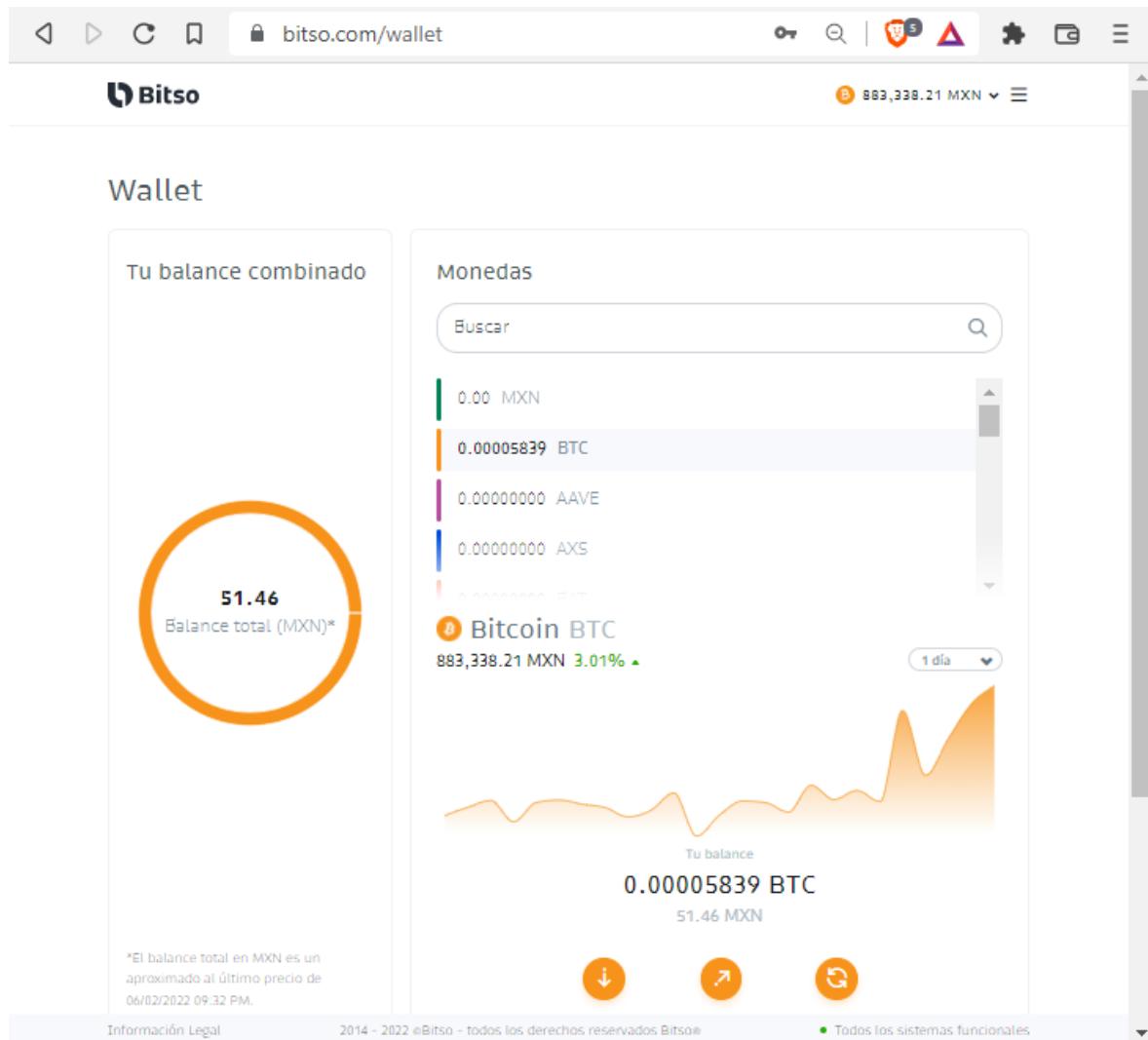
Envío de bitcoin: confirmación y notificación de operación



6. Por otro lado, el usuario proveedor del bien o servicio también recibirá una notificación de la operación por correo electrónico.

Figura 5-18*Recepción de bitcoin: notificación de operación*

7. Finalmente, una vez completadas las 4 confirmaciones requeridas por el Blockchain, los fondos serán acreditados y el usuario podrá ver su nuevo balance de bitcoin en su Wallet.

Figura 5-19*Recepción de bitcoin: nuevo balance en Wallet*

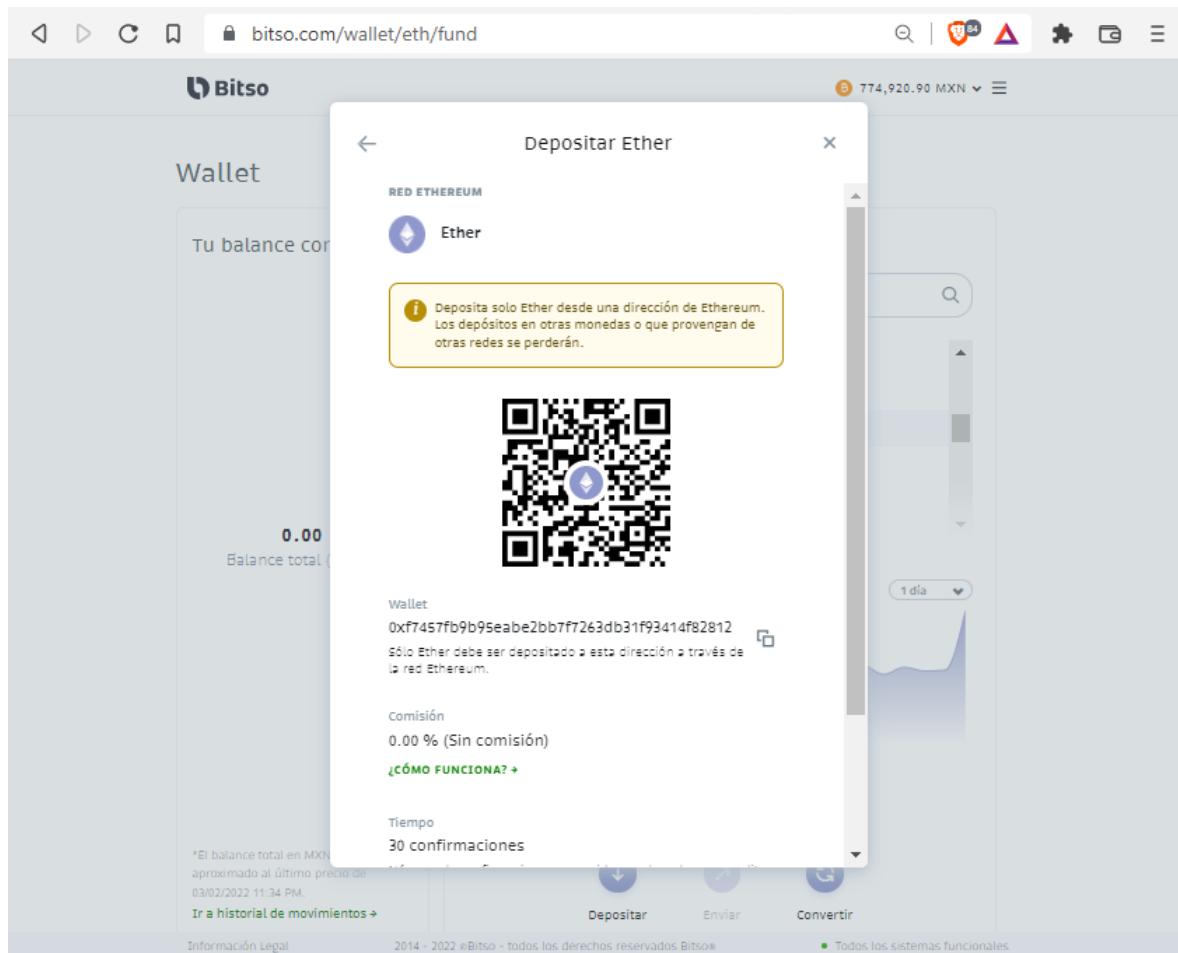
5.3.4 ¿Cómo recibir y realizar un pago con otra criptomoneda?

Como se comentó previamente, los procedimientos para recibir y realizar un pago con otra criptomoneda serán muy similares a los revisados previamente con bitcoin, aunque con ligeras variaciones dependiendo de la criptomoneda en cuestión o el exchange utilizado.

Para recibir un pago con una criptomoneda diferente a bitcoin, se seguirán los mismos pasos utilizados en la sección “5.3.2.2 Dirección bitcoin dinámica”. La principal diferencia es que para la mayoría de los casos, la dirección obtenida sería estática en lugar de dinámica, como es el caso de la criptomoneda ether (ETH).

Figura 5-20

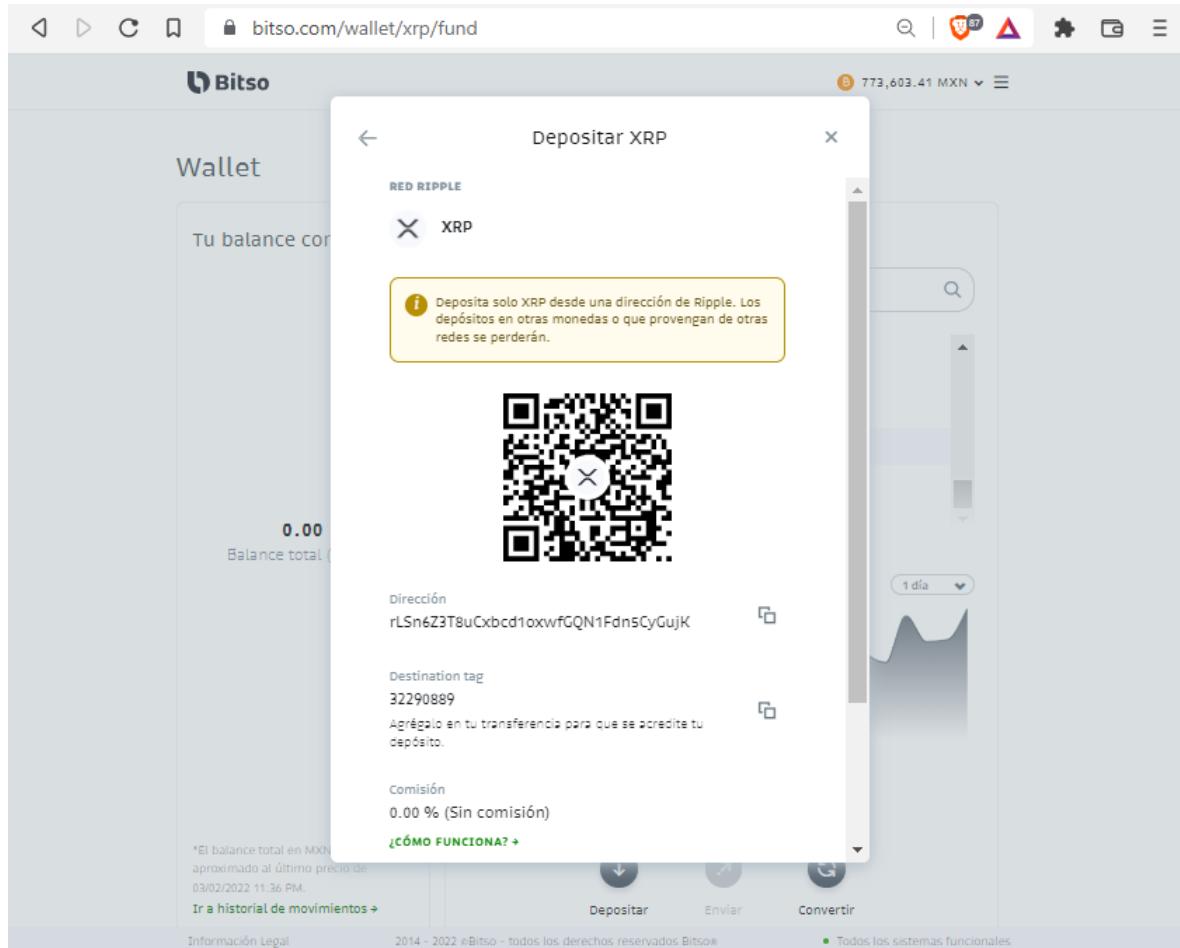
Bitso: ejemplo de código QR y dirección ether (ETH)



Por otro lado, para realizar un pago con una criptomoneda diferente a bitcoin, se seguirán los mismos pasos utilizados en la sección “5.3.3 ¿Cómo realizar un pago con bitcoin?”. Las diferencias más importantes que hay que tomar en cuenta son que habrá que elegir la red Blockchain adecuada para la criptomoneda en cuestión y verificar si esta requiere algún parámetro o identificador adicional, como es el caso para la criptomoneda ripple (XRP).

Figura 5-21

Bitso: ejemplo de código QR, destination tag y dirección ripple (XRP)

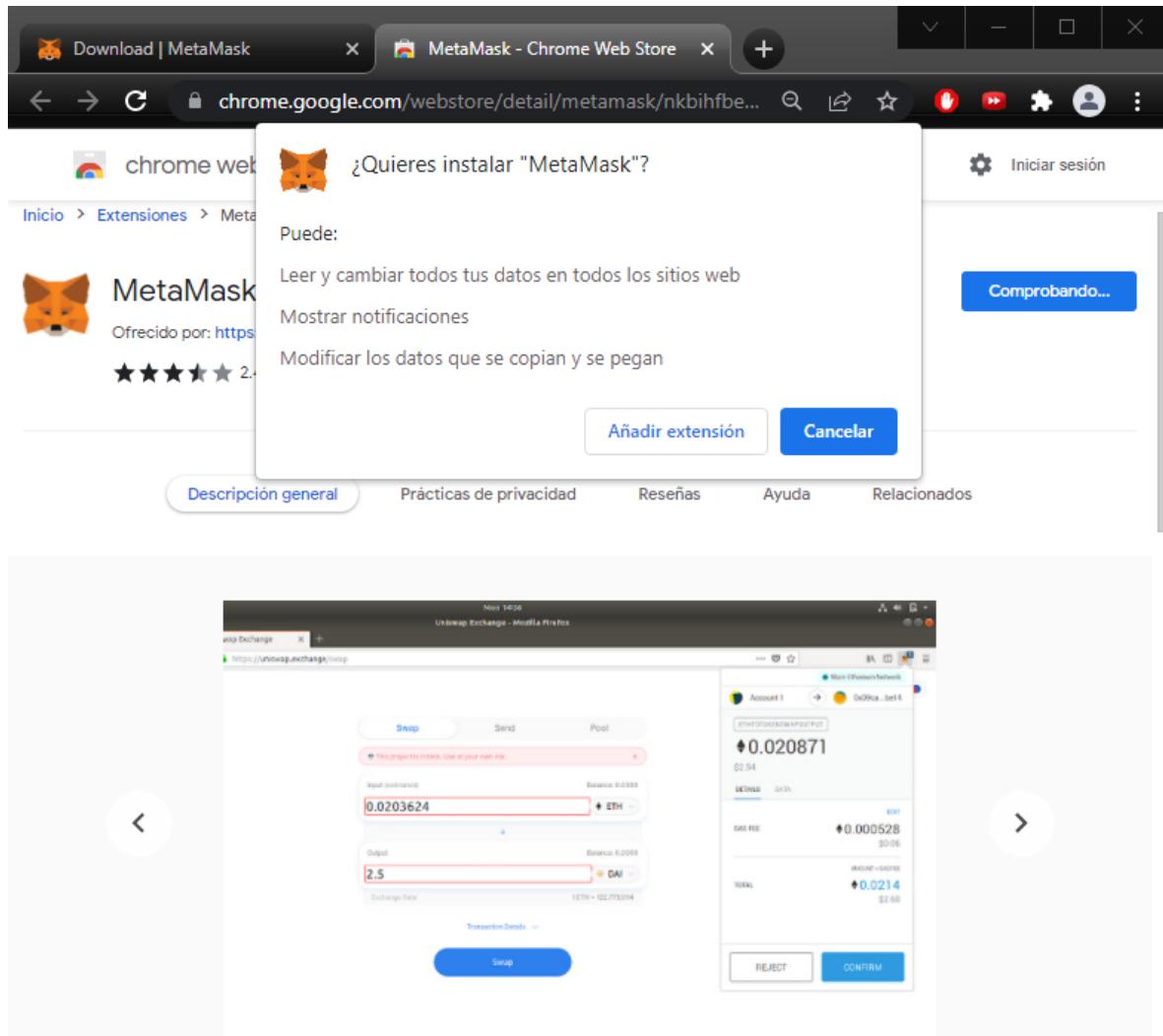


5.4 Prueba de concepto para contrato inteligente

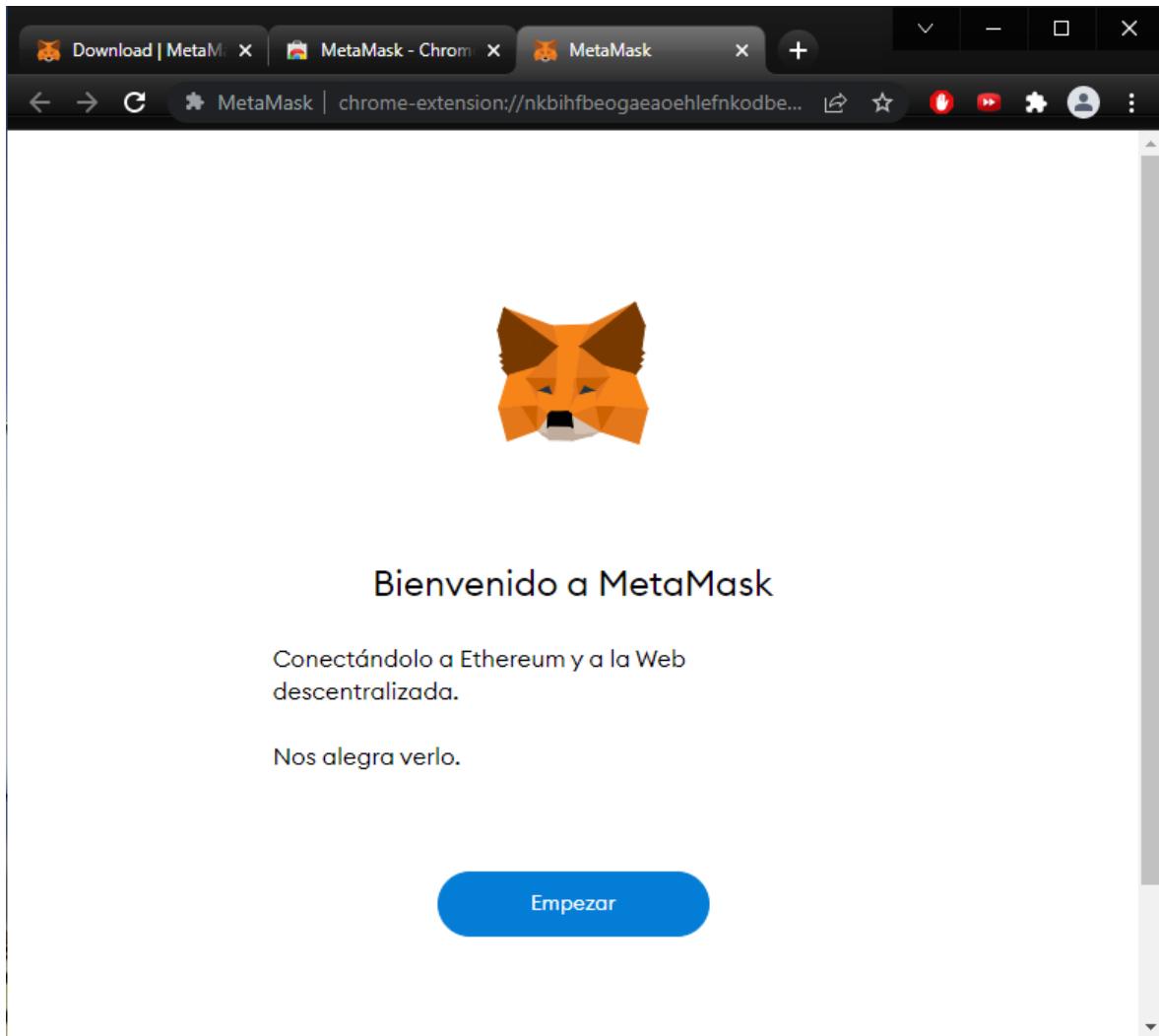
5.4.1 Configuración de Wallet MetaMask

Para poder interactuar con smart contracts en el Blockchain de Ethereum a través de una de sus redes disponibles, el usuario deberá configurar la extensión de navegador del MetaMask Wallet, realizando los siguientes pasos:

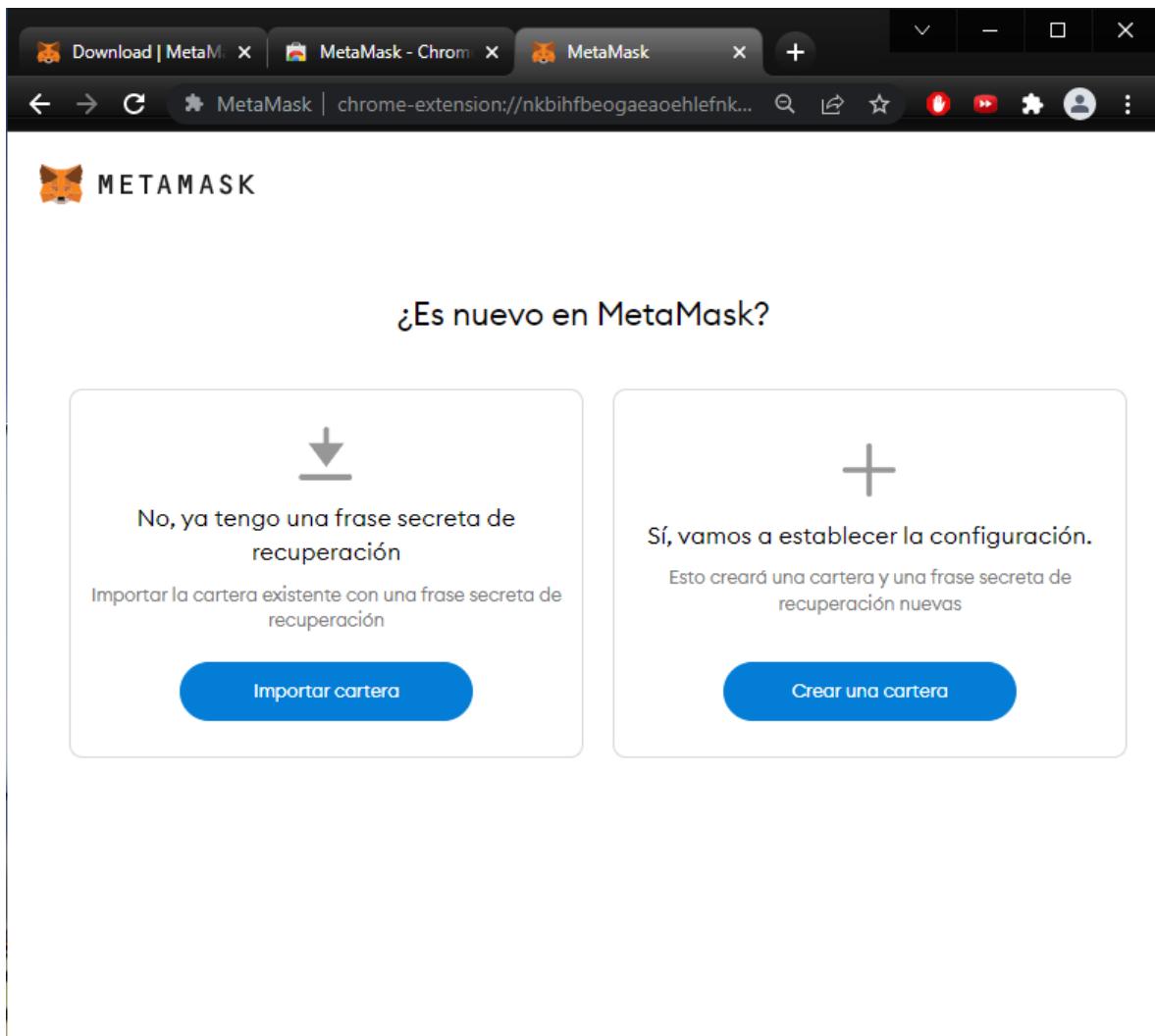
1. En principio, se deberá descargar e iniciar el asistente de instalación de la extensión de navegador, para este caso Google Chrome.

Figura 5-22*MetaMask: instalación de extensión en Google Chrome*

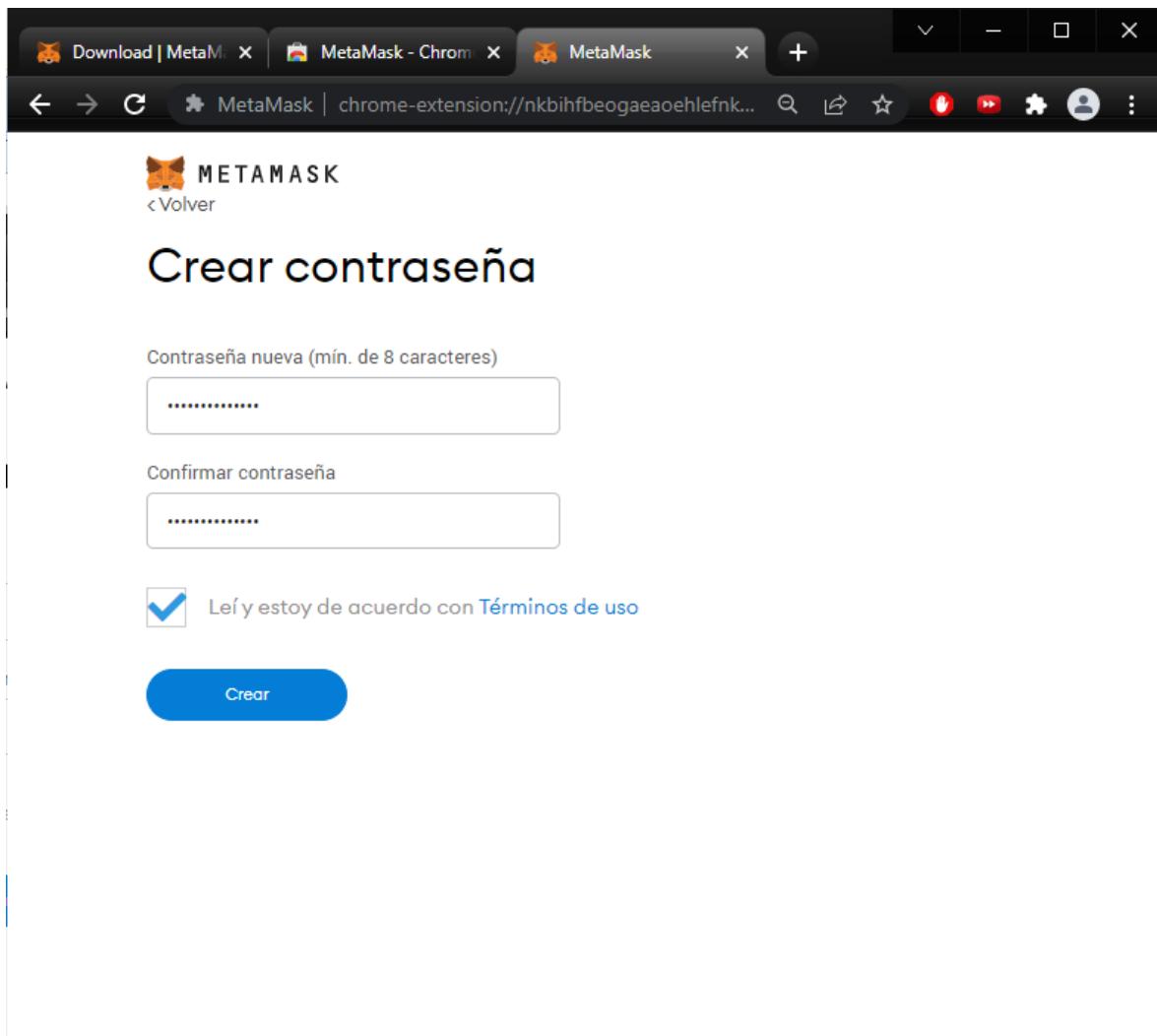
2. Una vez que la extensión fue instalada, aparecerá la pantalla de bienvenida del asistente de configuración.

Figura 5-23*MetaMask: pantalla de bienvenida*

3. El siguiente paso permitirá la importación de un Wallet existente a través de una frase secreta de recuperación, o bien la creación de uno nuevo. Para este ejemplo, se elegirá la segunda opción.

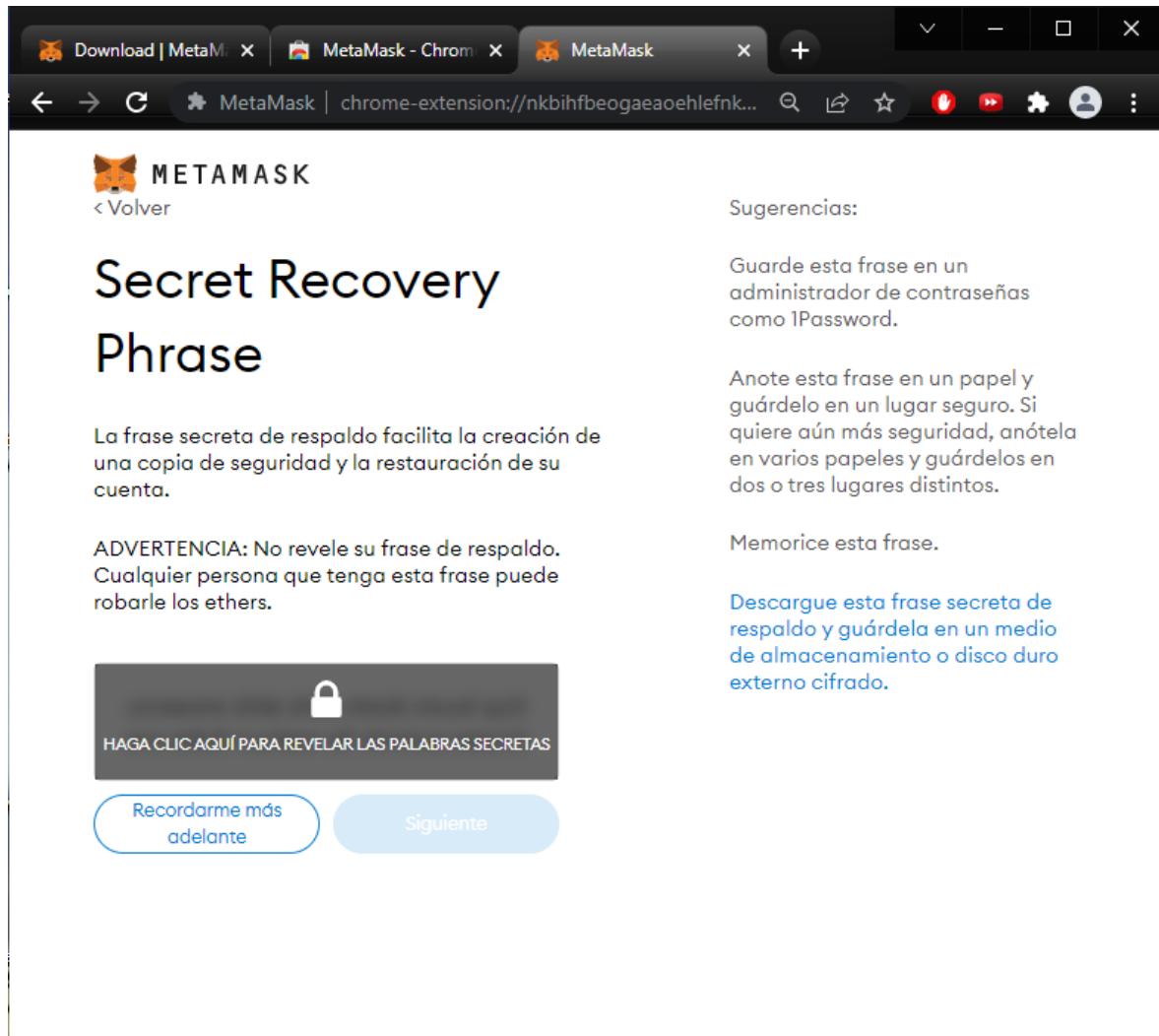
Figura 5-24*MetaMask: importación o creación de Wallet*

4. A continuación se tendrá que definir y confirmar una contraseña para acceder al Wallet.

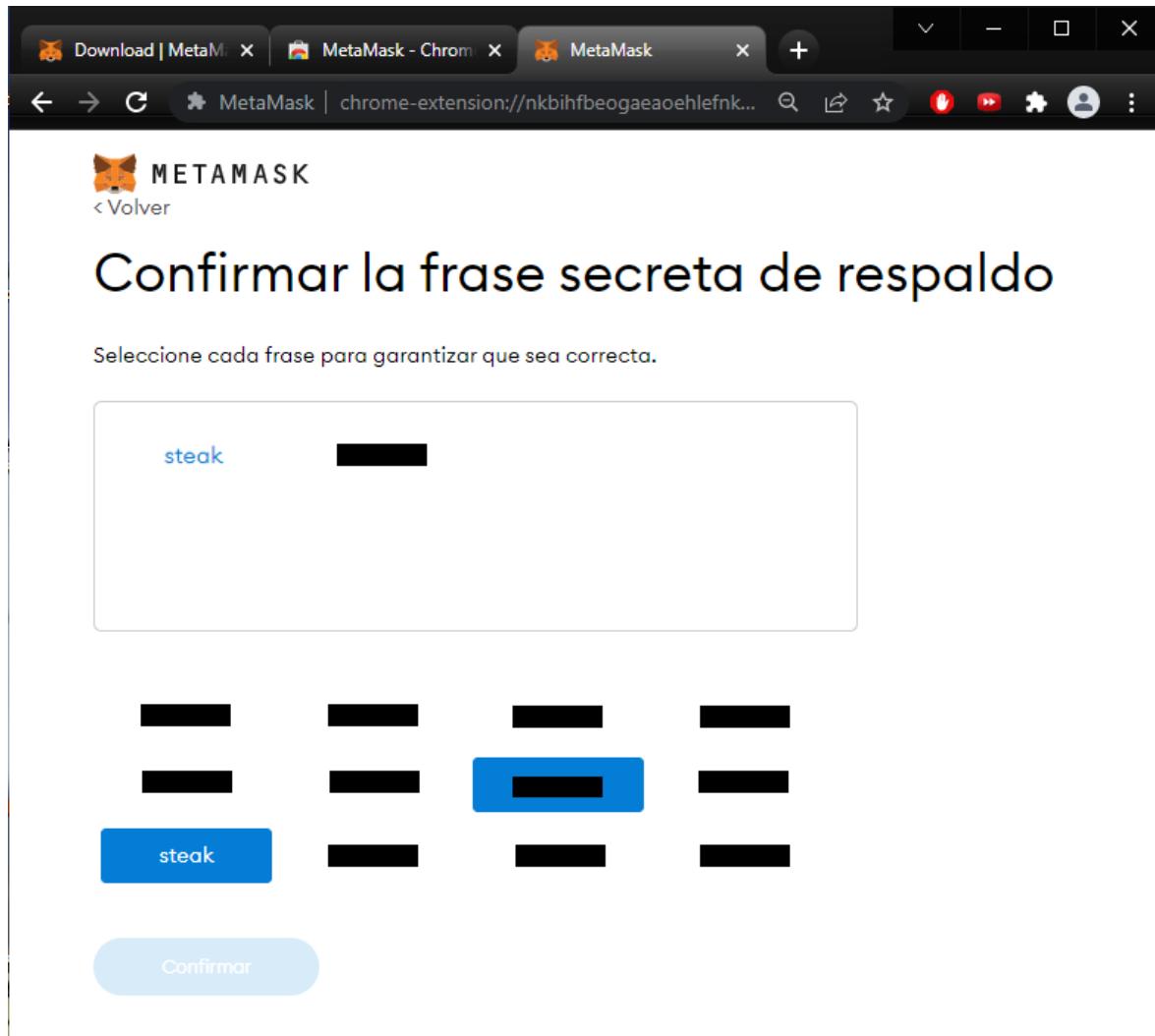
Figura 5-25*MetaMask: definición de contraseña*

5. Posteriormente, el asistente generará automáticamente la frase secreta de recuperación para el usuario, la cual consiste en 16 palabras que permitirán la restauración de la cuenta.

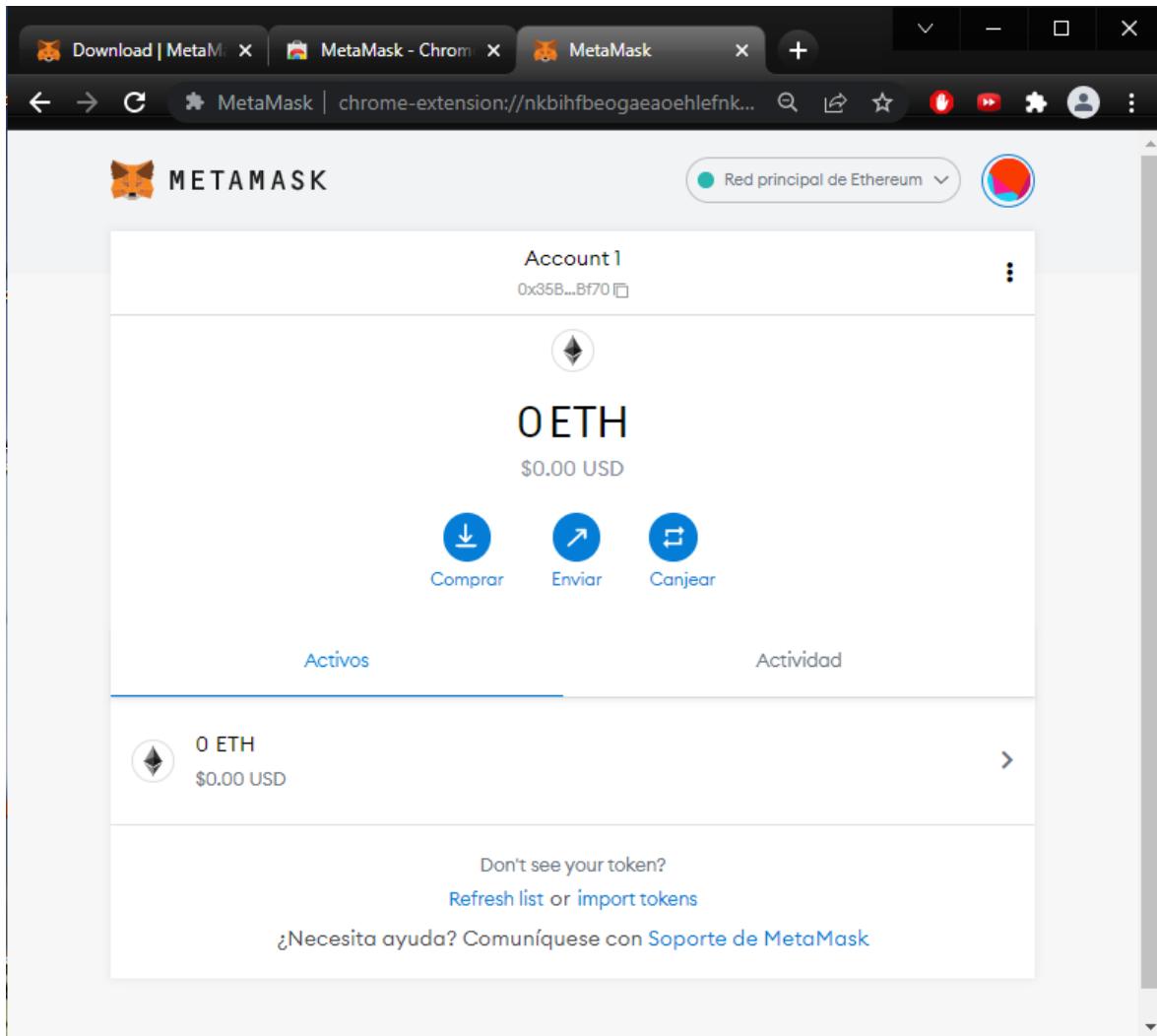
Esta frase debe permanecer en un lugar seguro y no ser compartida, ya que la persona que tuviera acceso a ella podría disponer de todos los recursos asociados a la cuenta. Por otro lado, Metamask tampoco puede recuperar una frase secreta.

Figura 5-26*MetaMask: generación de frase secreta de recuperación*

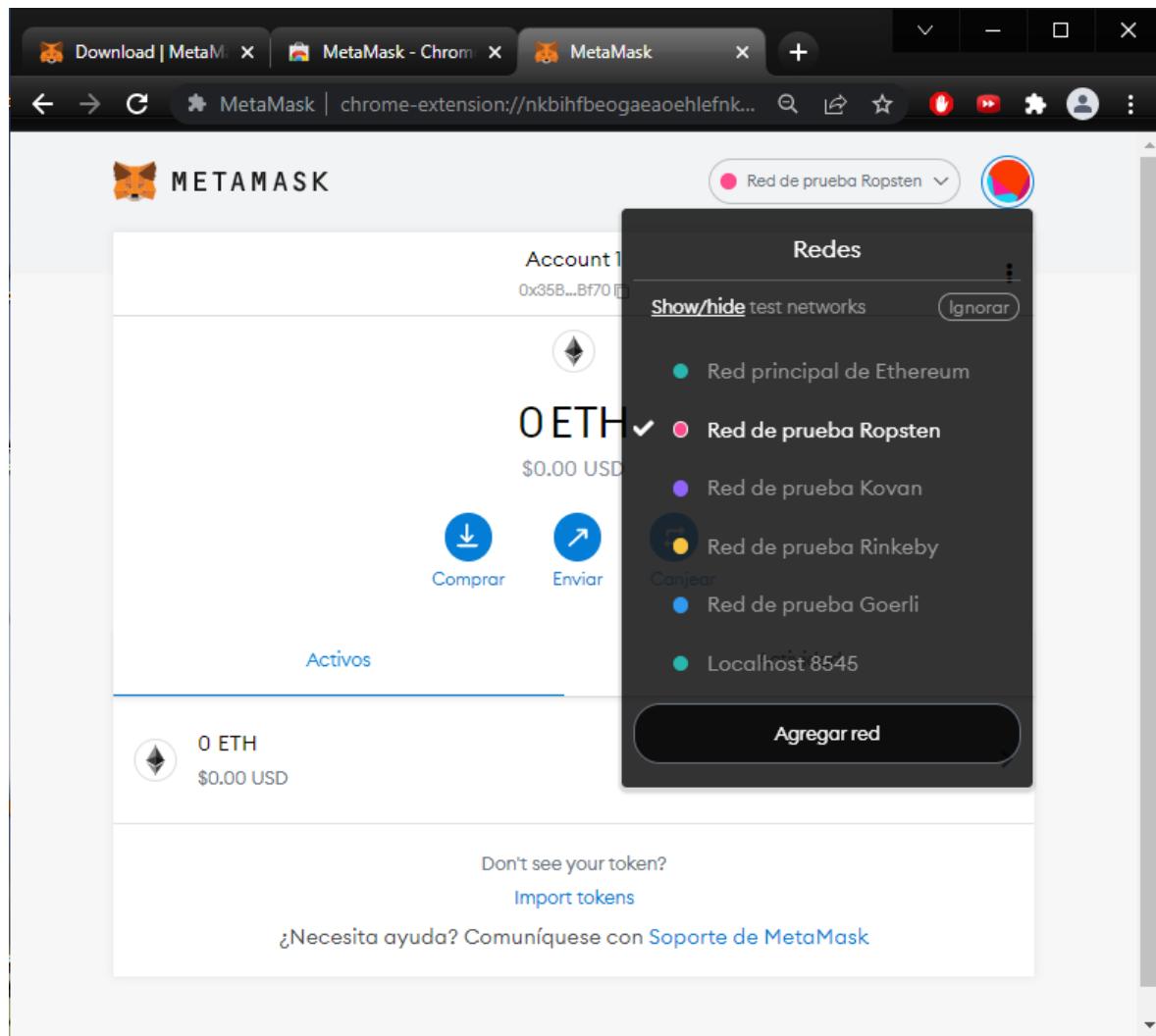
6. El siguiente paso consistirá en confirmar la frase secreta autogenerada del paso anterior, y para ello se deberán seleccionar las palabras que la forman en el orden correcto.

Figura 5-27*MetaMask: confirmación de frase secreta de recuperación*

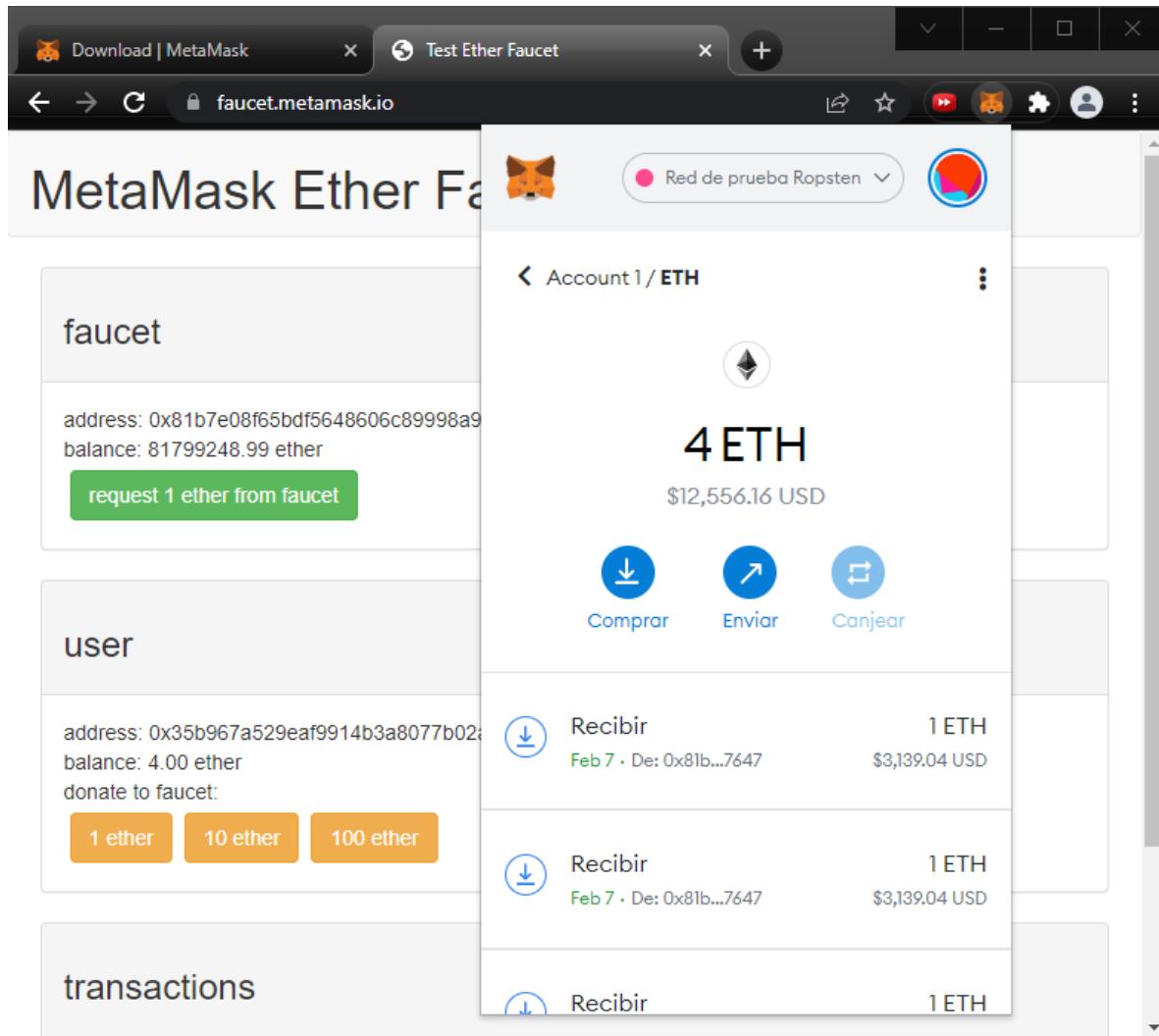
7. Con estos pasos el Wallet quedará configurado, por lo que tras seleccionar el ícono de la extensión MetaMask e introducir la contraseña adecuada para accederla, se mostrará la dirección asociada a la cuenta, además de su balance de ether en la red principal de Ethereum (Mainnet), el cual será de 0 ETH.

Figura 5-28*MetaMask: balance de Wallet en la red principal*

8. Acto seguido, se procederá a elegir la red de prueba (Testnet) Ropsten, la cual utiliza el mecanismo de Proof of Work, emulando el comportamiento de la red principal de Ethereum.

Figura 5-29*MetaMask: selección de red de prueba Ropsten*

9. Finalmente, se accederá a alguno de los sitios ether faucet disponibles en línea como <https://faucet.metamask.io/>, donde desde su apartado faucet, se introducirá la dirección ether de la cuenta para poder obtener 1 test ether por cada solicitud enviada, la cual se reflejará posteriormente en el balance de Wallet de la red de prueba Ropsten.

Figura 5-30*Obtención de test ether para red de prueba Ropsten y balance de Wallet*

5.4.2 Entorno de desarrollo Ethereum Remix IDE

Para crear un smart contract se necesitará de un ambiente de desarrollo apropiado con varias herramientas de Software por instalar.

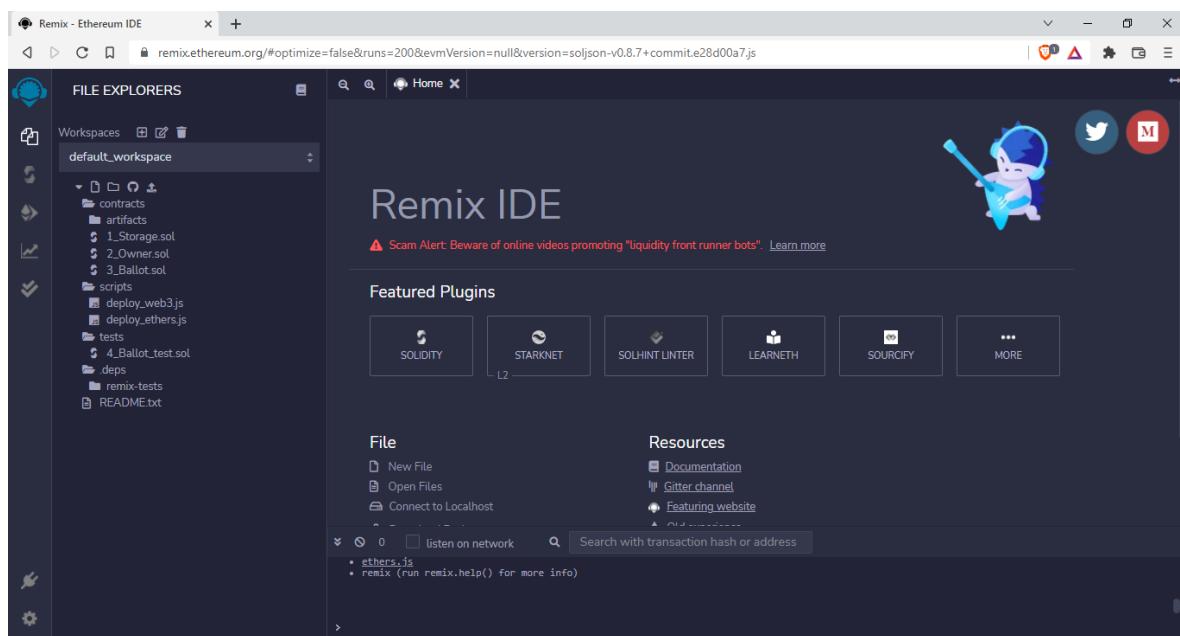
Para agilizar este proceso, en este ejemplo se utilizará el Ethereum Remix IDE disponible en <https://remix.ethereum.org/>, el cual es un entorno de desarrollo en

línea que proporcionará todas las herramientas necesarias para la creación, compilación y despliegue de un smart contract en el Blockchain de Ethereum.

En primer lugar y por defecto, al acceder a este IDE se mostrará la sección de explorador de archivos, la cual incluye 3 contratos sencillos precargados a manera de ejemplos, un área de trabajo para visualizar el código desarrollado, un área para mostrar los logs de ejecución y un menú lateral desde donde se tendrá acceso a otras secciones. Cada una de ellas tiene un propósito diferente dentro del IDE y estas son: explorador de archivos, compilador Solidity, despliegue y ejecución de transacciones, análisis estático y pruebas unitarias, siendo las tres primeras las que se utilizarán durante el desarrollo de la prueba de concepto.

Figura 5-31

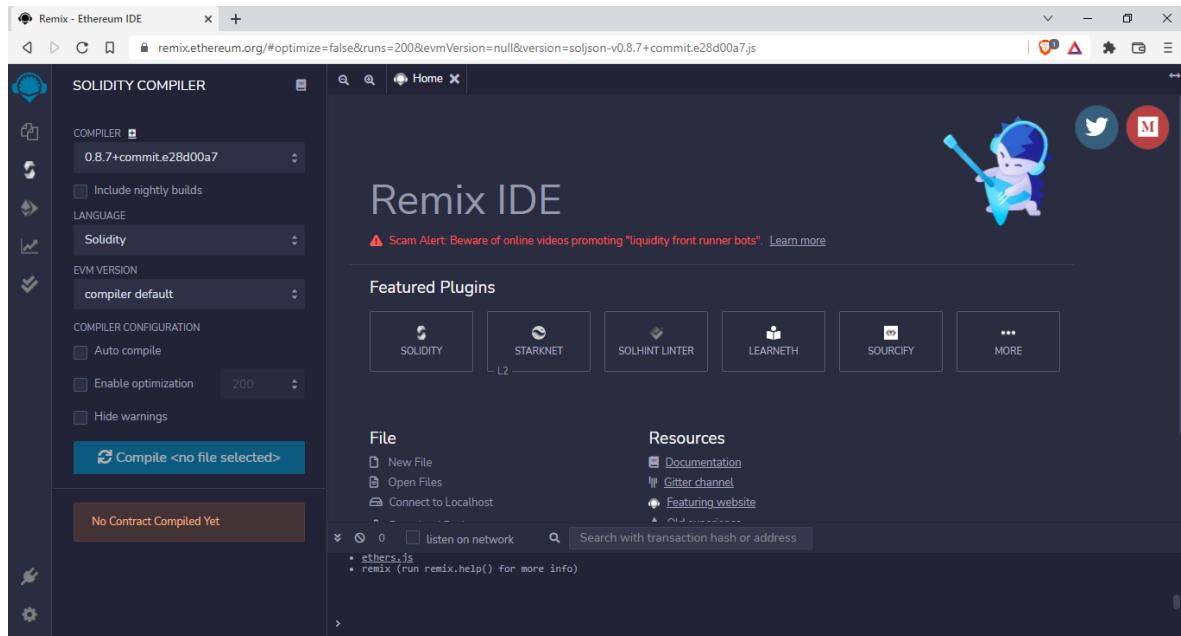
Remix IDE: explorador de archivos



En la sección de compilador Solidity, se podrán observar unas listas desplegables que permitirán elegir el lenguaje de programación a utilizar (Solidity), la versión adecuada del compilador y la versión de la EVM a utilizar, además de un botón para compilar el código del contrato mostrado en el área de trabajo.

Figura 5-32

Remix IDE: compilador Solidity



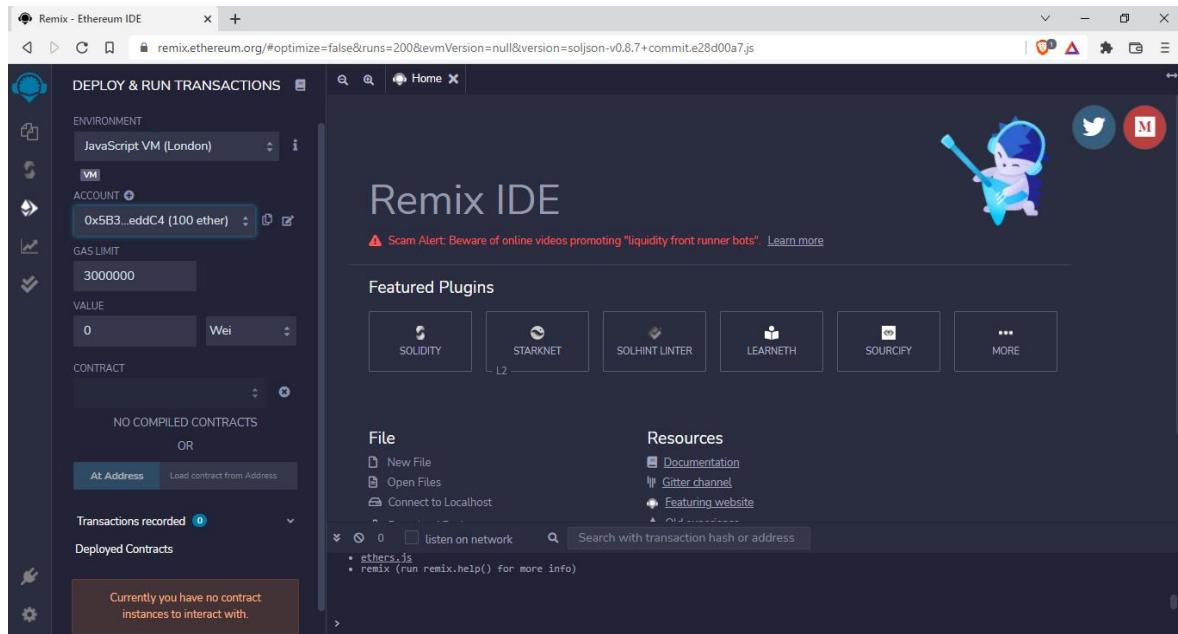
Finalmente, en la sección de despliegue y ejecución de transacciones, se destacan tres listas desplegables. En la primera de ellas se podrá elegir el ambiente donde se desplegará el smart contract, ya sea en una emulación de Blockchain con la opción Javascript VM, en un Blockchain privado mediante la opción Web3 Provider, o bien en una de las redes Ethereum a través de la opción Injected Web3. Para las dos últimas opciones, la autenticación del usuario se llevará a cabo mediante la extensión MetaMask.

En la segunda lista desplegable, se elegirá la dirección de la cuenta que realizará el despliegue del contrato y en la tercera, se seleccionará el nombre del contrato a desplegar, seguido del botón para realizar esta acción.

Por último, en seguida de los elementos anteriores, se mostrarán las cajas de texto y botones a manera de interfaz para poder interactuar con las funciones definidas en el smart contract, una vez que este haya sido correctamente desplegado.

Figura 5-33

Remix IDE: despliegue y ejecución de transacciones



5.4.3 Definición de caso de uso para Smart Contract

El escenario que se plantea como caso de uso genérico para esta prueba de concepto de smart contract, consiste en crear el registro permanente de algún objeto y los cambios en su estado, además de también poder ser transferido hacia otros usuarios, como podría ser el caso de automóviles, bienes raíces, obras de arte físicas o digitales, entre otros.

Ahora bien, aterrizando la idea hacia una implementación más concreta y de características similares, se propone la creación de un contrato para un evento de adopción de mascotas, donde por cada evento habrá un administrador y una lista con las diferentes mascotas que se encuentren en adopción, cada una con sus características particulares. Algunas de estas características podrán ser actualizadas con el tiempo, como el nombre o la edad de la mascota, si esta ha sido esterilizada, o también podría cambiar de dueño, por una u otra razón.

5.4.4 Codificación y compilación del Smart Contract

Una vez definido el escenario de implementación, se procederá a codificar sus requerimientos y características en el lenguaje de programación Solidity, a través del Remix IDE.

Cabe aclarar que la explicación de la sintaxis y semántica del lenguaje de programación Solidity está fuera del alcance de esta investigación, por lo que se deberá de consultar los enlaces pertinentes incluidos en la bibliografía.

Continuando con el ejemplo, una vez en la sección de explorador de archivos del IDE, dentro del directorio contracts se creará un nuevo archivo llamado EventoAdopcion.sol, al cual se le copiará el siguiente código fuente:

```
// SPDX-License-Identifier: GPL-3.0
// Author: Gerardo Cataño
// version: 0.1
pragma solidity >=0.7.0 <0.9.0;

/* Prueba de concepto de smart contract para el registro de mascotas
   y su posterior seguimiento en un evento de adopcion */
contract EventoAdopcion {

    //modelo de datos de la mascota
    struct Mascota {
        uint8 id;
        address dueno;          //propiedad actualizable
        string nombre;          //propiedad actualizable
        string especie;
        string raza;
        string genero;
        string color;
        uint8 edad;              //propiedad actualizable
        bool esterilizada;      //propiedad actualizable
    }

    //se define al responsable del evento
    address administradorEvento = msg.sender;
```

```

//se define lista y contador de mascotas
mapping(uint8 => Mascota) public mascotas;
uint8 public contadorMascotas;

//se añade la lista de mascotas al inicializar el evento
constructor () {
    anadirMascotaEnAdopcion("Tito", "perro", "schnauzer", "macho", "gris plata", 9);
    anadirMascotaEnAdopcion("Pelusa", "gato", "persa", "hembra", "beige", 5);
    anadirMascotaEnAdopcion("Layla", "perro", "chihuahua", "hembra", "negro", 6);
    anadirMascotaEnAdopcion("Morgan", "gato", "criollo", "macho", "blanco", 2);
}

//funcion interna de apoyo para inicializar lista y contador de mascotas
function anadirMascotaEnAdopcion(string memory _nombre, string memory _especie, string memory _raza,
                                    string memory _genero, string memory _color, uint8 _edad) private {
    contadorMascotas++;
    mascotas[contadorMascotas] =
        Mascota(contadorMascotas, address(0), _nombre, _especie, _raza, _genero, _color, _edad, false);
}

//funcion interna de apoyo para validar que el identificador de la mascota
// este dentro del rango de la lista
function validaIdentificadorMascota(uint8 _idMascota) private view {
    require(mascotas[_idMascota].id >= 1 && mascotas[_idMascota].id <= contadorMascotas,
            "El identificador de mascota introducido no esta registrado");
}

//funcion interna de apoyo para validar que solo el administrador o el
// dueño de la mascota puedan realizar una operacion del contrato
function validaAdministradorODueno(uint8 _idMascota) private view {
    if (_msg.sender != administradorEvento && _msg.sender != mascotas[_idMascota].dueno) {
        revert("Solo el administrador del evento o el dueño pueden realizar esta acción");
    }
}

//funcion publica para realizar el cambio de dueño de una mascota, tras haber validado requisitos
function cambiaDueñoMascota(uint8 _idMascota, address _nuevoDueño) public returns (string memory){
    validaIdentificadorMascota(_idMascota);
    validaAdministradorODueno(_idMascota);
    mascotas[_idMascota].dueno = _nuevoDueño;
    return "El dueño de la mascota fue actualizada";
}

```

```

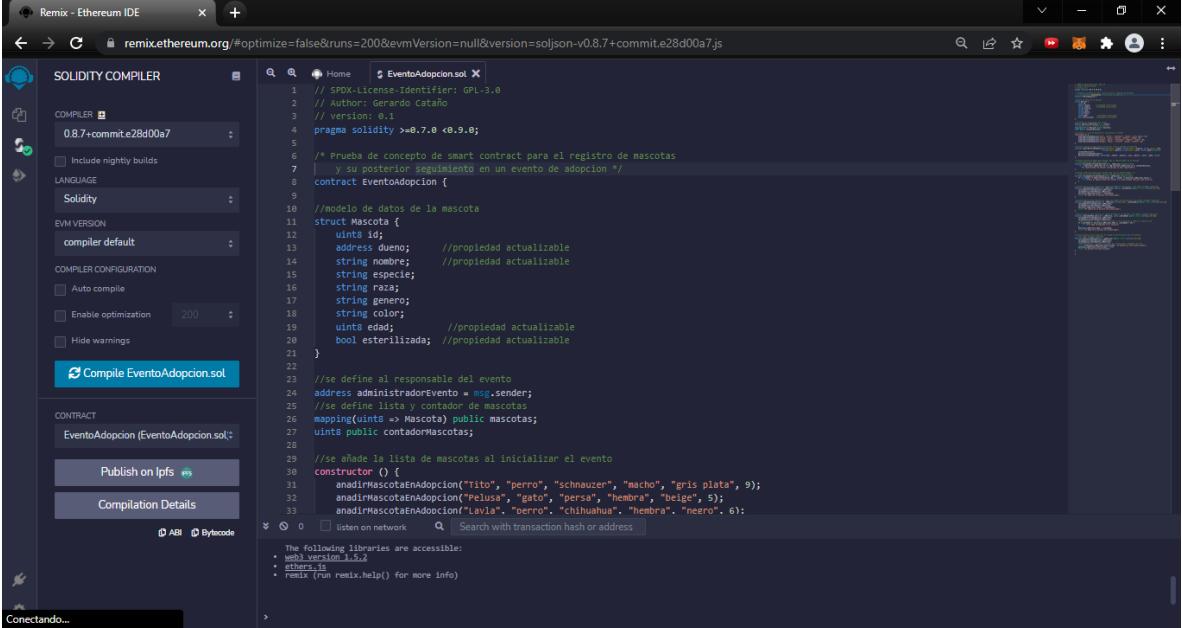
//funcion publica para realizar el cambio de nombre de una mascota, tras haber validado requisitos
function cambiaNombreMascota(uint8 _idMascota, string memory _nuevoNombre) public returns (string memory){
    validaIdentificadorMascota(_idMascota);
    validaAdministradorODueno(_idMascota);
    mascotas[_idMascota].nombre = _nuevoNombre;
    return "El nombre de la mascota fue actualizada";
}

//funcion publica para realizar el cambio de edad de una mascota, tras haber validado requisitos
function cambiaEdadMascota(uint8 _idMascota, uint8 _nuevaEdad) public returns (string memory){
    validaIdentificadorMascota(_idMascota);
    validaAdministradorODueno(_idMascota);
    //adicionalmente, la edad no debe ser menor a la ya definida y tendra un limite de 25
    if (_nuevaEdad <= mascotas[_idMascota].edad || _nuevaEdad > 25) {
        revert("La edad introducida no es valida");
    }
    mascotas[_idMascota].edad = _nuevaEdad;
    return "La edad de la mascota fue actualizada";
}

//funcion publica para actualizar el estado de esterilizacion de una mascota,
// tras haber validado requisitos
function esterilizaMascota(uint8 _idMascota) public returns (string memory){
    validaIdentificadorMascota(_idMascota);
    validaAdministradorODueno(_idMascota);
    //adicionalmente, el estado solo podra ser actualizado a verdadero una vez
    require(!mascotas[_idMascota].esterilizada, "La mascota ya esta esterilizada");
    mascotas[_idMascota].esterilizada = true;
    return "La mascota fue esterilizada";
}
}

```

A continuación, dentro de la sección del compilador de Solidity, se elegirán las versiones de lenguaje y EVM por defecto, se seleccionará el contrato EventoAdopcion y se procederá a hacer click en el botón correspondiente para compilar el archivo EventoAdopcion.sol.

Figura 5-34*Compilación del contrato EventoAdopcion.sol*


```

SPDX-License-Identifier: GPL-3.0
// Author: Gerardo Cataño
// version: 0.1
pragma solidity >=0.7.0 <0.9.0;

/* Prueba de concepto de smart contract para el registro de mascotas
   y su posterior seguimiento en un evento de adopción */
contract EventoAdopcion {

    //Modelo de datos de la mascota
    struct Mascota {
        uint id;
        address dueno;      //propiedad actualizable
        string nombre;      //propiedad actualizable
        string especie;
        string raza;
        string genero;
        string color;
        uint8 edad;         //propiedad actualizable
        bool esterilizada; //propiedad actualizable
    }

    //se define al responsable del evento
    address administradorevento = msg.sender;
    //se define lista y contador de mascotas
    mapping(uint => Mascota) public mascotas;
    uint public contadormascotas;
}

//se añade la lista de mascotas al inicializar el evento
constructor () {
    anadirMascotaEnAdopcion("Tito", "perro", "schnauzer", "macho", "gris plata", 9);
    anadirMascotaEnAdopcion("Felusa", "gato", "persa", "hembra", "beige", 5);
    anadirMascotaEnAdopcion("Lalí", "orro", "chihuahua", "hembra", "neero", 6);
}

```

The screenshot shows the Remix Ethereum IDE interface. On the left, the Solidity Compiler settings are shown, including the compiler version (0.8.7+commit.e28d00a7), language (Solidity), EVM version (compiler default), and compiler configuration (Auto compile). The right pane displays the source code of the `EventoAdopcion.sol` contract. The code defines a `Mascota` struct with fields for ID, owner, name, species, breed, gender, color, age, and sterilization status. It also defines a mapping from a uint index to a `Mascota` object, a counter for the number of pets, and a constructor that adds several pre-defined pets to the mapping.

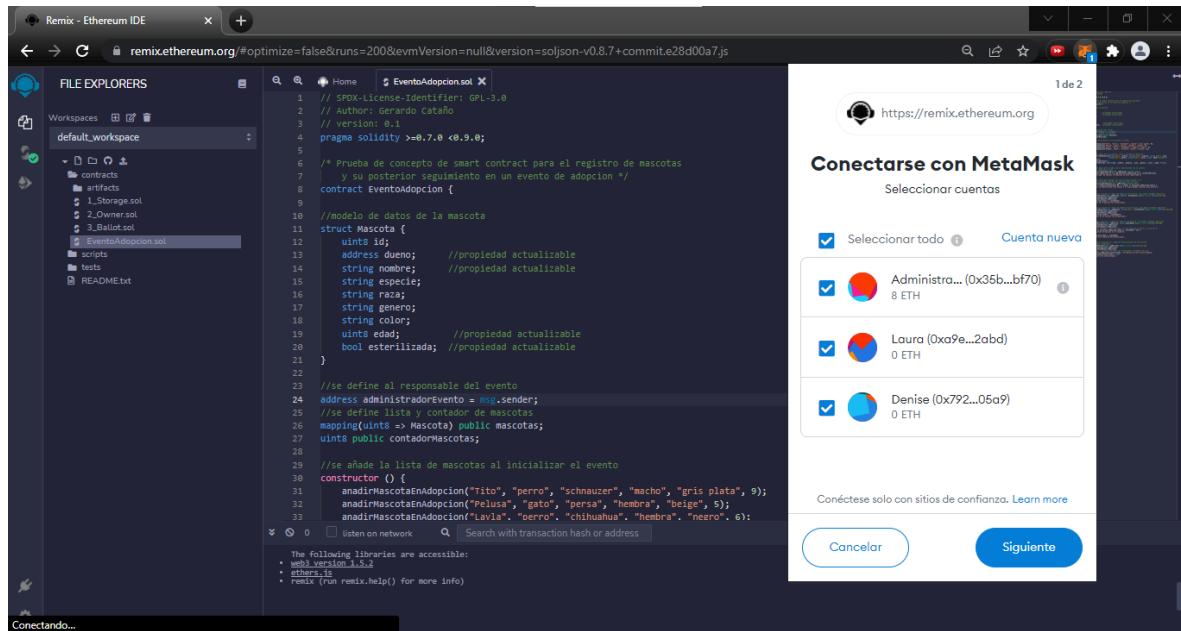
5.4.5 Despliegue del Smart Contract

Es importante señalar que durante las etapas de codificación y pruebas, es recomendable utilizar el ambiente JavaScript VM y las direcciones de prueba disponibles con esta opción, sin embargo para desplegar el contrato y realizar transacciones dentro de la red de prueba Ropsten en el Blockchain de Ethereum, será necesario configurar el cliente MetaMask.

Para ello, se registrarán un par de cuentas adicionales, se les asignará un nombre diferente al genérico, se les transferirá saldo en test ether y por último a cada una se le permitirá conectar con el sitio de la IDE: <https://remix.ethereum.org/>.

Figura 5-35

Conexión del Remix IDE con MetaMask



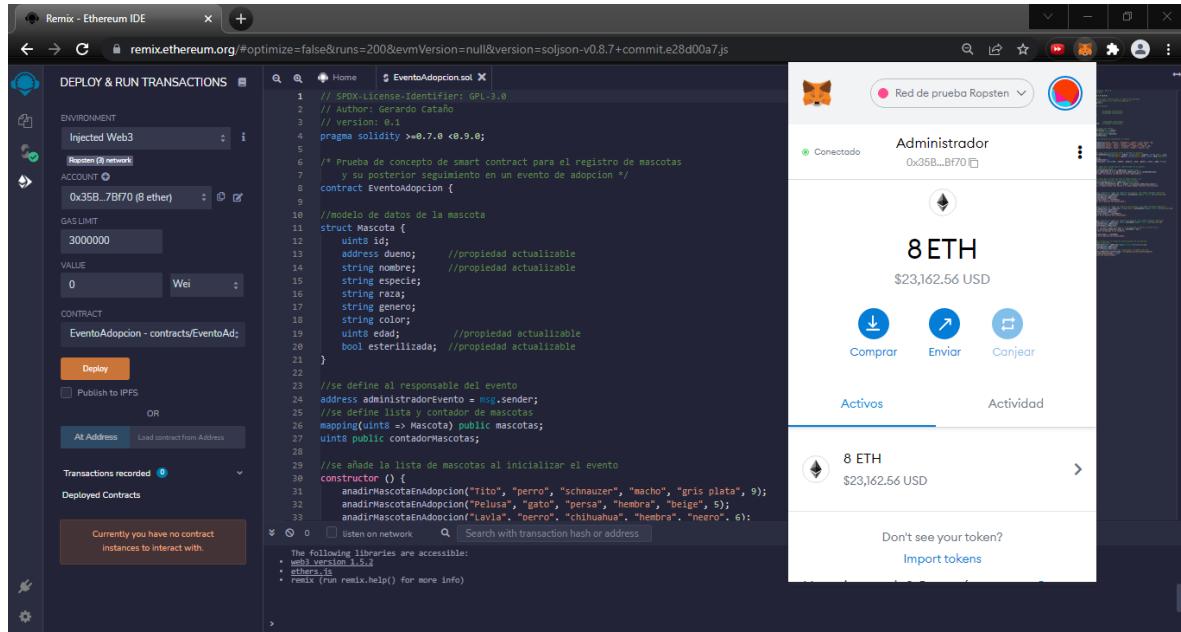
Una vez completado este paso, se accederá a la sección de despliegue y ejecución de transacciones. En esta sección, desde la lista desplegable de ambientes se elegirá la opción Injected Web3, lo cual mostrará la dirección que actualmente se encuentre conectada con MetaMask, en la lista desplegable inferior.

Para este caso, se elegirá la cuenta en MetaMask desde donde se desee realizar la transacción para el despliegue del smart contract.

A continuación, se seleccionará el nombre del contrato previamente compilado desde la lista desplegable correspondiente y se procederá a hacer click en el botón de despliegue.

Figura 5-36

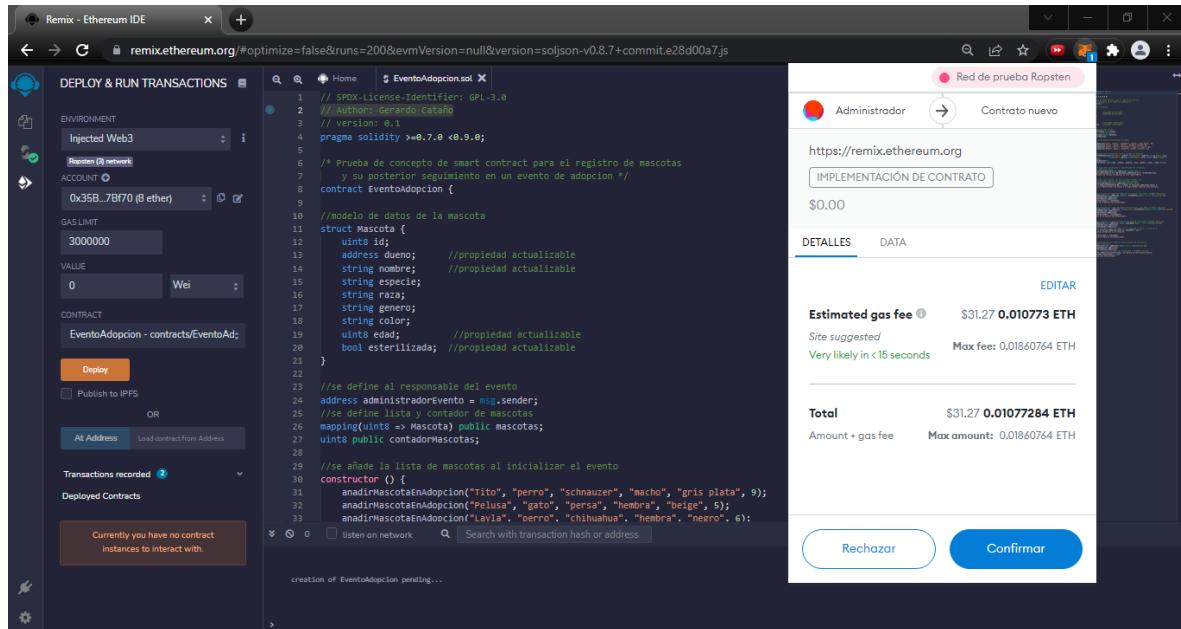
Opciones de configuración para despliegue del Smart Contract



Esta acción quedará asentada en el log de transacciones del IDE y el cliente MetaMask mostrará el Gas estimado para la creación y despliegue del contrato.

Figura 5-37

Gas estimado para despliegue del Smart Contract

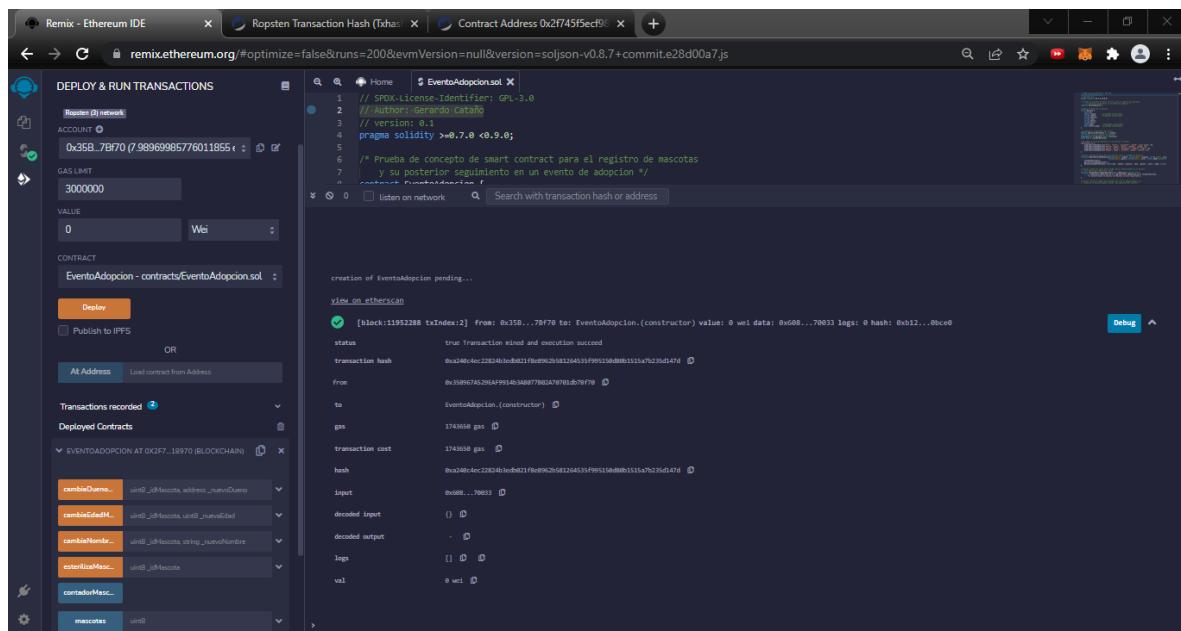


Posteriormente, tras haber revisado las cuotas para el despliegue del contrato, se procederá a confirmar la transacción. Con esta acción, se desplegará el contrato en la red Ropsten del Blockchain de Ethereum y los detalles de la transacción, así como su enlace directo en el sitio <https://ropsten.etherscan.io/>, se podrán observar en el log de transacciones del IDE.

Finalmente, en el apartado de contratos desplegados del IDE aparecerá una interfaz sencilla compuesta de botones y cajas de texto, elementos que permitirán la interacción con las funciones que fueron definidas en el contrato.

Figura 5-38

Confirmación de despliegue del Smart Contract e interfaz para su utilización



5.4.6 Interacción con el Smart Contract

Para utilizar las funciones definidas en el contrato, es importante conocer que existen dos tipos de interacciones: las de tipo call y las de tipo transact.

Las interacciones de tipo transact implican el cambio de estado de un elemento en el Blockchain, lo que se traduce en el costo del Gas consumido durante la ejecución de las operaciones que la conformen. Estas se representan en la interfaz del IDE con botones de color naranja.

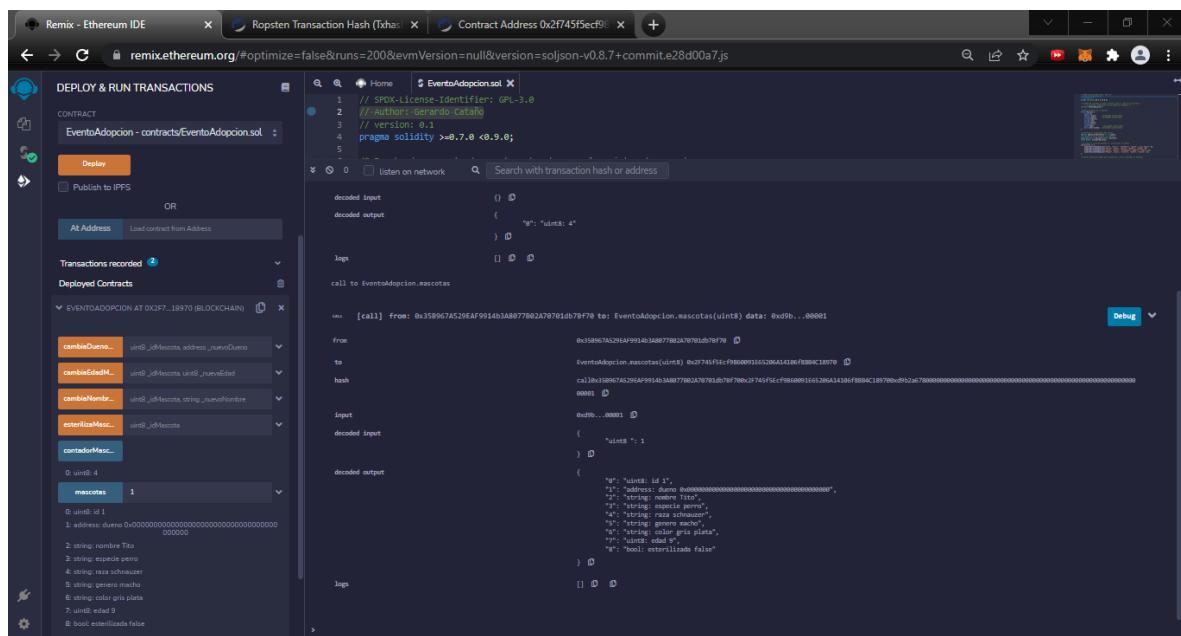
Por otra parte, las interacciones de tipo call no tienen ningún costo, ya que están formadas por operaciones de solo lectura; además de estar representados con botones de color azul en la interfaz del IDE.

Para comenzar con las pruebas de interacción con el contrato, se procederá a realizar las interacciones tipo call. Primero se hará click en el botón *contadorMascotas*, el cual devolverá un valor de 4, lo que representa el número de mascotas registradas para este escenario.

Posteriormente, se introducirá el valor de “1” en la caja de texto asociada al botón *mascotas*, y posteriormente se le hará click. Esto mostrará todas las propiedades y sus respectivos valores asociados a la mascota con ese identificador.

Figura 5-39

Interacciones de tipo call



Como se mencionó anteriormente, ambas interacciones tipo call se ejecutarán inmediatamente sin necesidad de interactuar con MetaMask y quedarán registradas en el log del IDE, más no dentro del Blockchain.

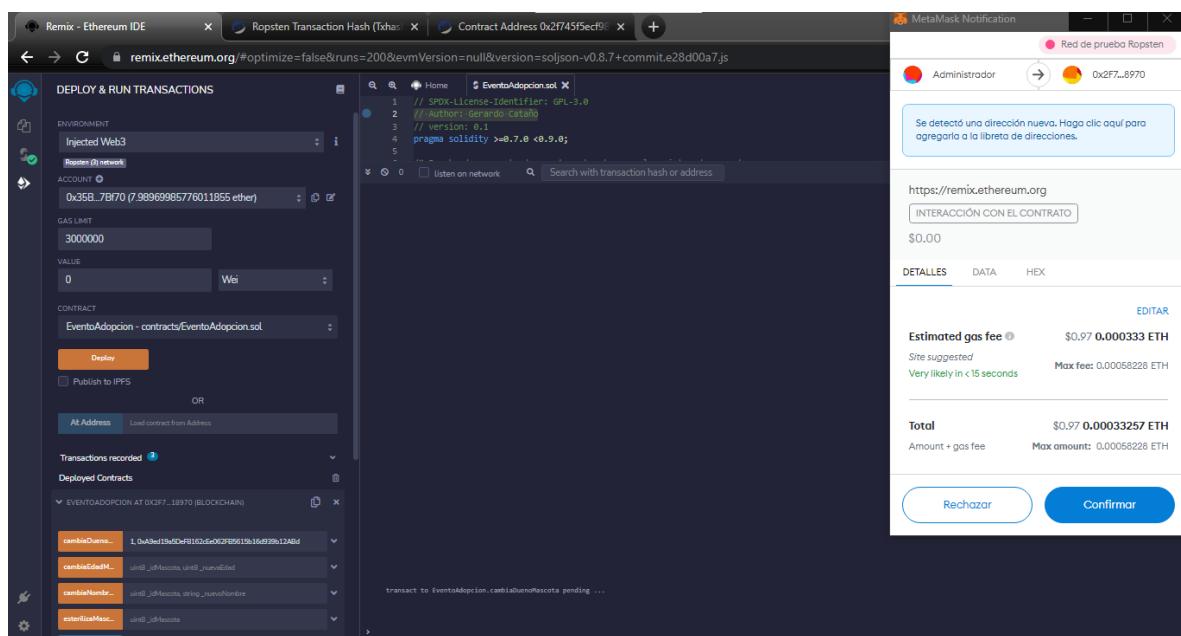
Continuando con las pruebas de interacción con el contrato, se procederá a realizar solo un par de interacciones del tipo transact, ya que los pasos a seguir para interactuar con cada son muy similares.

Primeramente, utilizando la cuenta asociada como administrador del evento (0x35B967A529EAF9914b3A8077B02A70701db7Bf70), se introducirán los valores “1, 0xA9ed19a5DeF8162cEe062FB5615b16d939b12ABd” en la caja de texto asociada al botón *cambiaDuenoMascota* y se le dará click, lo cual significará asignar la dirección del nuevo dueño a la mascota con identificador 1.

Como se mencionó previamente, las interacciones tipo transact involucrarán un costo, por lo requerirán ser confirmadas desde MetaMask.

Figura 5-40

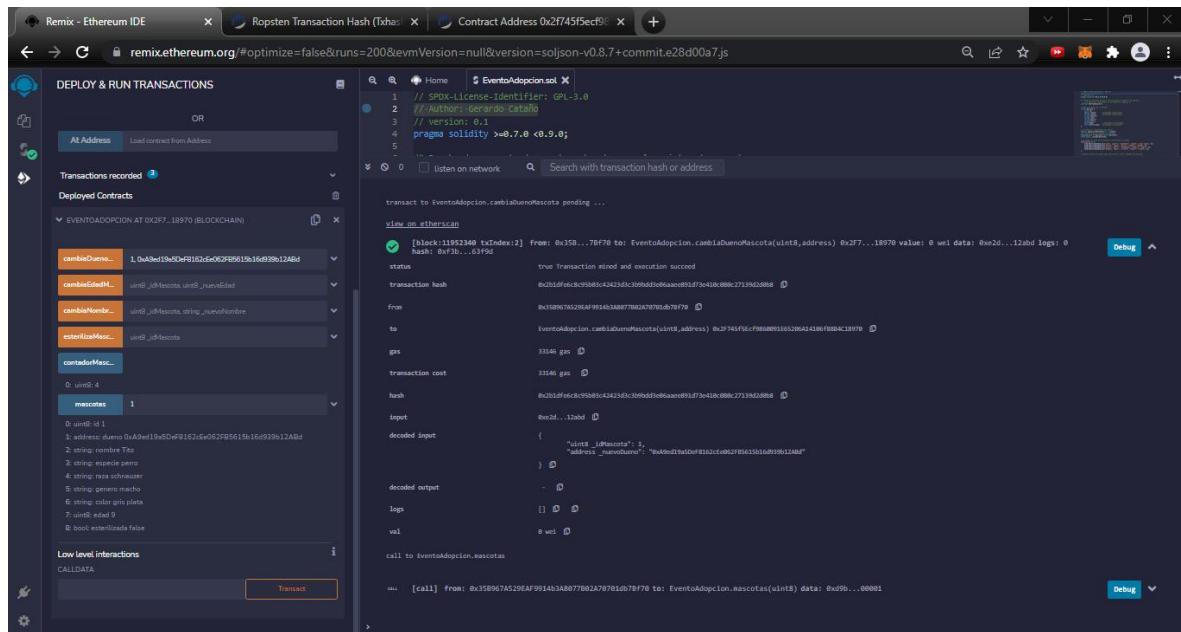
Interacción de tipo transact: función cambiaDuenoMascota



Una vez confirmada la transacción *cambiaDueñoMascota*, esta se registrará en el Blockchain y en log de transacciones del IDE, por lo que se tendrá que hacer una nueva llamada a la función *mascotas* con el valor “1”, para ver la propiedad *dueño* actualizada.

Figura 5-41

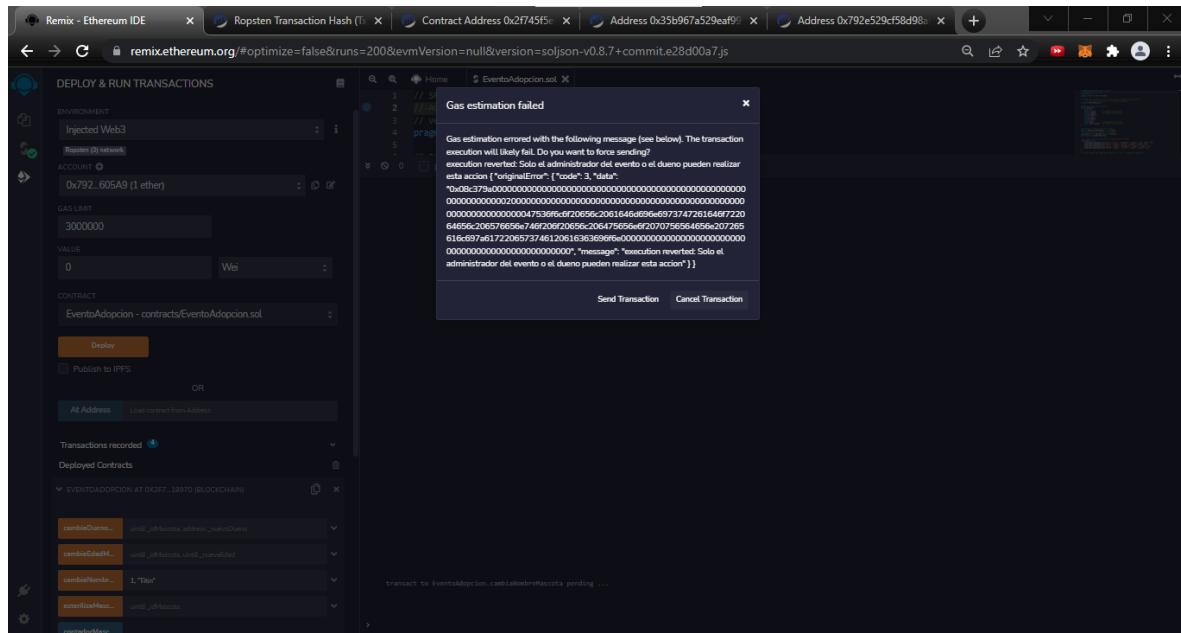
Transacción satisfactoria: *cambiaDueñoMascota*



Para la siguiente prueba, se utilizará la cuenta `0x792E529cf58D98A65ccd8a28677F379b891605A9`, y se introducirán los argumentos “1, “Titin”” para la función *cambiaNombreMascota*, haciendo click en el botón asociado. No obstante, debido a que no se cumplen los requerimientos de validación (la cuenta desde donde se llama la función no es el administrador del evento ni el dueño de la mascota), el IDE, a través de los cálculos de estimación de Gas, mostrará una advertencia indicando que de continuar con la solicitud la transacción fallará, por lo que se procederá a hacer click en el botón de cancelar transacción.

Figura 5-42

Advertencia de fallo de transacción cambiaNombreMascota desde IDE



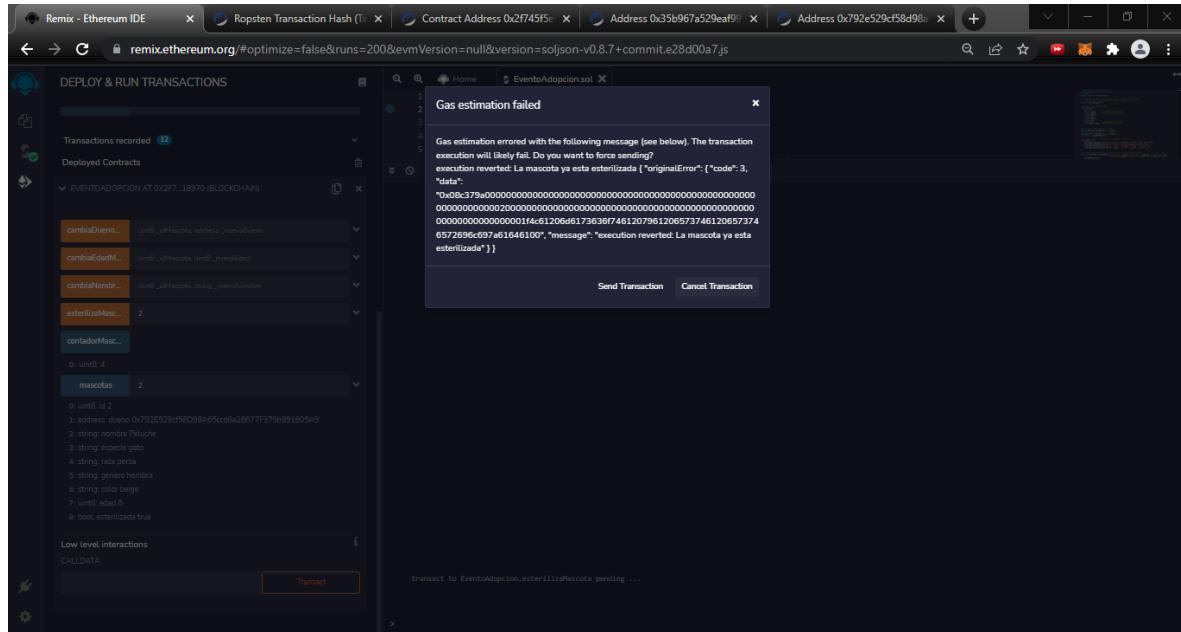
Acto seguido, desde la cuenta de administrador se procederá a asociar la mascota “2” a la cuenta 0x792E529cf58D98A65ccd8a28677F379b891605A9 a través de la función *cambiaDueñoMascota*.

Posteriormente, desde esta última cuenta 0x792E529cf58D98A65ccd8a28677F379b891605A9, se procederá a llamar secuencialmente las siguientes funciones con los respectivos parámetros especificados: *cambiaEdadMascota(2,6)*, *cambiaNombreMascota(2, “Peluche”)*, *esterilizaMascota(2)* y *mascotas(2)*, lo cual realizará actualizaciones en las propiedades de la mascota con identificador 2.

Para concluir, se repetirá la función *esterilizaMascota(2)* desde la cuenta 0x792E529cf58D98A65ccd8a28677F379b891605A9, lo cual implica un incumplimiento en los requerimientos de validación y por consecuencia, el IDE mostrará una nueva advertencia que de continuar con la solicitud, la transacción fallará.

Figura 5-43

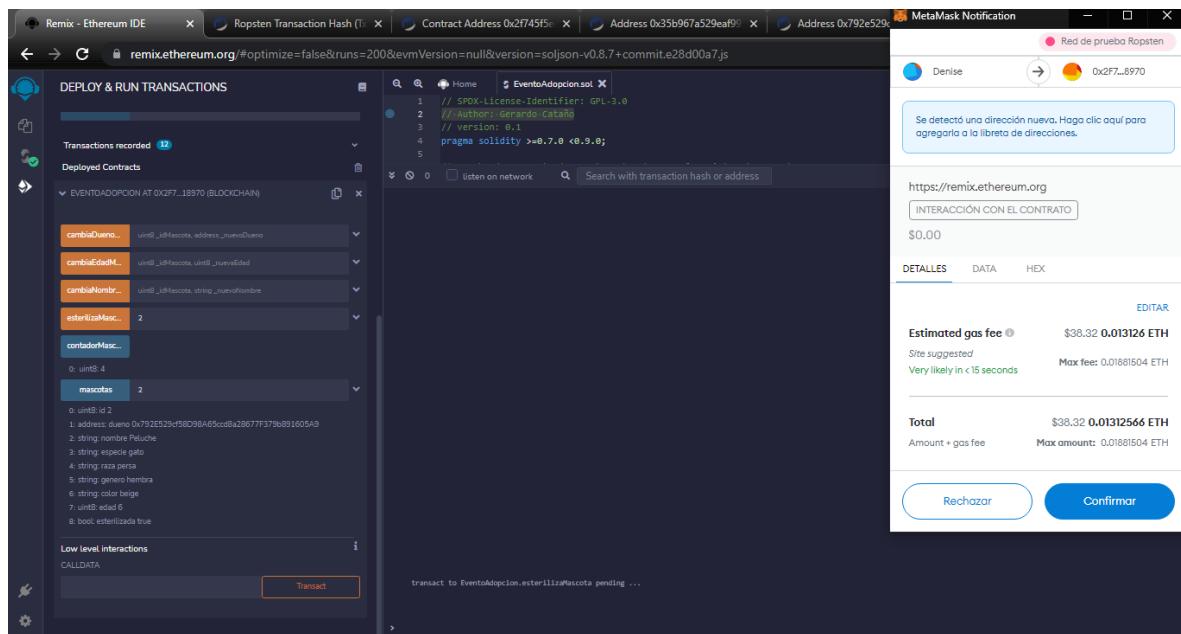
Advertencia de fallo de transacción esterilizaMascota desde IDE



No obstante, en esta ocasión se procederá a enviar la transacción a pesar de la advertencia y posteriormente, se confirmará la transacción en MetaMask.

Figura 5-44

Confirmación de transacción esterilizaMascota



Acto seguido, se podrá confirmar que la transacción falló, lo cual quedará registrado en el log de transacciones del IDE y en el Blockchain.

Figura 5-45

Transacción fallida: esterilizaMascota

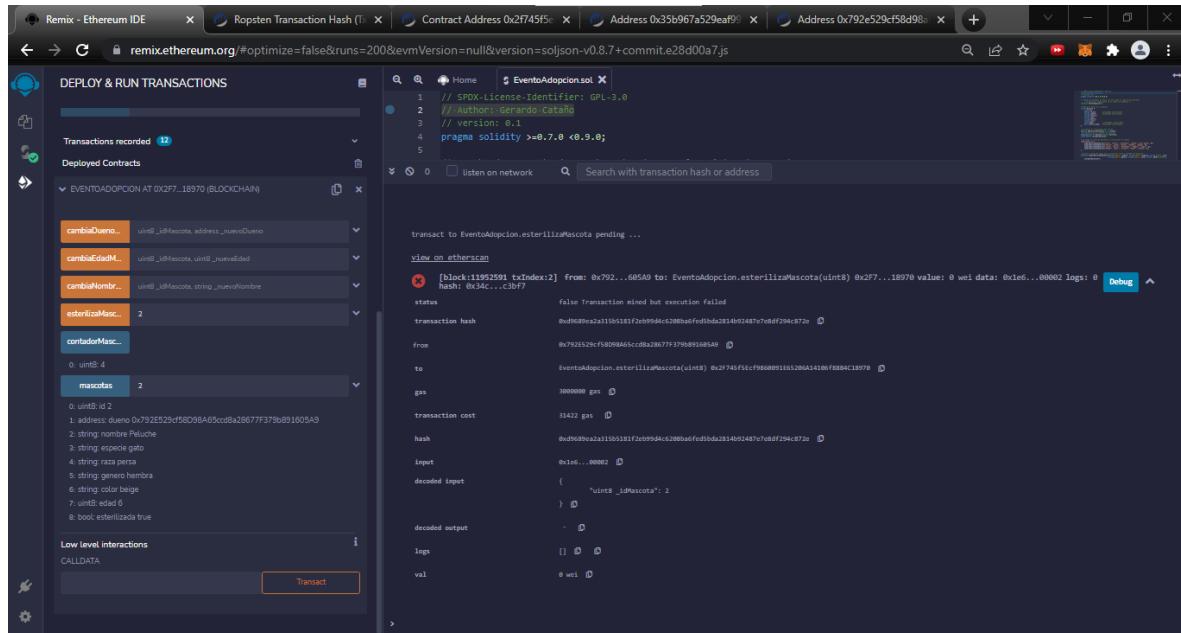
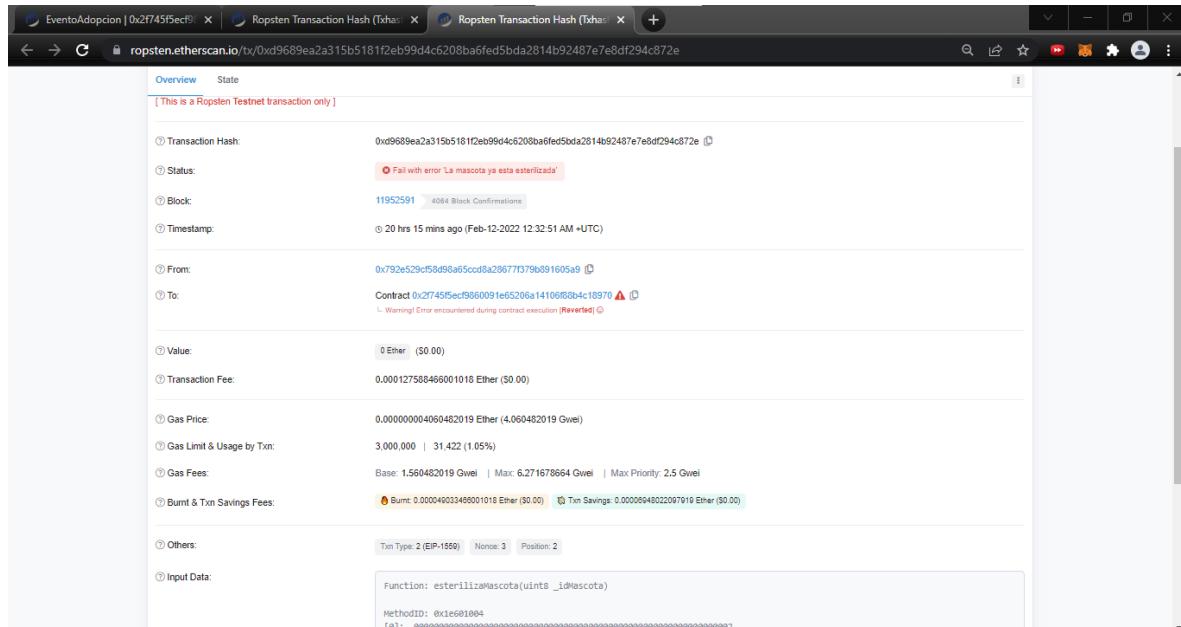


Figura 5-46

Transacción fallida: esterilizaMascota en Etherscan



Finalmente, cabe recalcar que todas las transacciones realizadas en el smart contract, así como las cuentas que interactuaron con este y sus respectivos históricos, podrán ser observables y verificables a través de un explorador de Blockchain, como es el caso de Etherscan para este ejemplo, donde la dirección asignada para el contrato fue:

<https://ropsten.etherscan.io/address/0x2f745f5ecf9860091e65206a14106f88b4c18970>.

Figura 5-47

Historial de transacciones realizadas en el Smart Contract

Txn Hash	Method	Block	Age	From	To	Value	Txn Fee
0xd9689ea2a31565181f...	Esteriliza Masc...	11952591	20 hrs 15 mins ago	0x792e529d58d98a65cc...	IN 0x2f745f5ecf9860091e6...	0 Ether	0.000127886466
0x6e9eb2a8e34d99c65...	Esteriliza Masc...	11952578	20 hrs 19 mins ago	0x792e529d58d98a65cc...	IN 0x2f745f5ecf9860091e6...	0 Ether	0.00014476453
0x0f9bc80595fa7cf641...	Cambia Nombre Ma...	11952568	20 hrs 21 mins ago	0x792e529d58d98a65cc...	IN 0x2f745f5ecf9860091e6...	0 Ether	0.000168402827
0x1d130561ba1ec6c36c...	Cambia Edad Masc...	11952553	20 hrs 24 mins ago	0x792e529d58d98a65cc...	IN 0x2f745f5ecf9860091e6...	0 Ether	0.000172837087
0x65b2c8097bc6eaf8ac0...	Cambia Dueño Mas...	11952531	20 hrs 30 mins ago	0x35b967a529eaf9914b...	IN 0x2f745f5ecf9860091e6...	0 Ether	0.000168977331
0x2b1dfe6c8c85b03c424...	Cambia Dueño Mas...	11952340	21 hrs 24 mins ago	0x35b967a529eaf9914b...	IN 0x2f745f5ecf9860091e6...	0 Ether	0.000313548299
0xa240c4ec22824b3edb...	[redacted]	11952288	21 hrs 42 mins ago	0x35b967a529eaf9914b...	IN [redacted] Create: EventoAdopcion	0 Ether	0.010300142239

6 Conclusiones

6.1 Conclusiones

El presente documento supone un inicio sencillo y práctico para aquella persona que esté interesada en comprender los conceptos generales de Blockchain y en conocer algunas alternativas para poder aplicarlos a su negocio.

En esta investigación, se hizo una recopilación y un estudio exhaustivo de material de diversas fuentes que fue revisado y comprendido para poder presentar los fundamentos de Blockchain, activos virtuales y contratos inteligentes de una manera más digerible para el lector.

También se elaboró una guía funcional que muestra el paso a paso de la adopción y la utilización de un método de pago con la criptomoneda bitcoin, orientada a cualquier MiPyMe o persona física en México. Esto se logró gracias a los servicios proporcionados por el exchange mexicano Bitso, donde se creó una cuenta lista para ser utilizada y sin tener comisiones por apertura o algún otro costo asociado.

Finalmente y de manera práctica, se configuraron los componentes de Software requeridos y se desarrolló un ejemplo de contrato inteligente, lo cual tampoco tuvo costo, ya que los componentes de Software utilizados son de código abierto y la implementación fue desplegada sobre la red de prueba Ropsten del Blockchain de Ethereum.

Este contrato inteligente sirve como base para la implementación de nuevas ideas en futuros desarrollos, en donde tendrá que ser adaptado a las necesidades específicas del negocio en cuestión y de cada caso de uso particular.

Los resultados fueron los esperados y no se presentaron mayores contratiempos durante el desarrollo de la investigación, aunque muchas veces hubo que ser selectivo con la información y corroborarla en fuentes más confiables.

6.2 Recomendaciones

Aunque la adopción de un mecanismo de pago con criptomonedas puede considerarse buena idea para la mayoría de los casos en escenarios de negocio, esto no siempre será aplicable para la implementación de un caso de uso con un contrato inteligente.

Por el contrario, esto solo sería factible en ciertos escenarios donde se tenga la necesidad de compartir información y confiar en terceros, además de que la información quede registrada y sea transparente para todos los involucrados, en tanto que aspectos como la privacidad de esta información no sea un asunto crítico.

Por este motivo, en la mayoría de los casos será recomendable partir de la necesidades específicas del negocio, planear una estrategia adecuada apoyándose de las diferentes áreas implicadas y después evaluar si es viable o no la implementación técnica de algún caso de uso a través de un contrato inteligente.

6.3 Trabajos futuros

De cara a nuevos estudios, sería conveniente aprender conceptos más avanzados de las tecnologías revisadas de manera general en esta investigación, así como conocer acerca de las particularidades, ventajas y beneficios que otras plataformas consolidadas de Blockchain ofrecen.

Las tecnologías Blockchain forman la infraestructura sobre la cual se seguirán construyendo nuevas capacidades y servicios que emulan el funcionamiento de aplicaciones tradicionales bajo esta nueva arquitectura. Estas han ido surgiendo gracias al desarrollo de aplicaciones descentralizadas, convirtiendo esta actividad en un área de investigación prometedora.

Actualmente, existen ideas novedosas como los NFTs (Non-Fungible Tokens), que promueven la venta de activos digitales como imágenes, fotografías, audios o videos mientras mantienen su autenticidad, unicidad y se protegen sus derechos de propiedad intelectual, o conceptos más innovadores como el Metaverso, el cual se ha ido mostrado como una combinación de varias áreas como la realidad virtual y aumentada, la inteligencia artificial, las redes sociales, videoconferencias, entre otras, que interactúan entre sí para brindar al usuario nuevas experiencias de intercomunicación, colaboración e inmersión en mundos digitales.

Ambos conceptos están relacionadas en mayor o menor medida con las tecnologías Blockchain, por lo que son líneas de investigación interesantes a las que bien valdría la pena seguirles la pista.

Referencias

Bashir, I. (2017). *Mastering Blockchain*.

Packt Publishing Ltd.

Bit2Me. (s.f.). *¿Qué es un exchange de criptomonedas?* Bit2Me Academy.

Recuperado el 4 de febrero de 2022, de

<https://academy.bit2me.com/que-es-exchange-criptomonedas/>

Gupta, M. (2017). *Blockchain For Dummies, IBM Limited Edition*.

John Wiley & Sons, Inc.

Haunts, S. (18 de septiembre de 2018b). *State of Blockchain: Executive Briefing*

[Curso en línea]. Pluralsight.

<https://app.pluralsight.com/library/courses/state-of-blockchain-executive-briefing>

Kolakowski, M. (9 de diciembre de 2021). *Ethereum Upgrade Delays 'Difficulty Bomb'*. Investopedia.

<https://www.investopedia.com/ethereum-upgrade-delays-difficulty-bomb-5212447>

National Institute of Standards and Technology (s.f.). Blockchain. En *Computer Security Resource Center's Glossary*. Recuperado el 2 de febrero de 2022, de

<https://csrc.nist.gov/glossary/term/blockchain>

National Institute of Standards and Technology (s.f.). Cryptocurrency. En *Computer Security Resource Center's Glossary*. Recuperado el 2 de febrero de 2022, de

<https://csrc.nist.gov/glossary/term/cryptocurrency>

National Institute of Standards and Technology (s.f.). Hash. En *Computer Security Resource Center's Glossary*. Recuperado el 2 de febrero de 2022, de

<https://csrc.nist.gov/glossary/term/hash>

National Institute of Standards and Technology (s.f.). Smart Contract. En *Computer Security Resource Center's Glossary*. Recuperado el 2 de febrero de 2022, de

https://csrc.nist.gov/glossary/term/smart_contract

National Institute of Standards and Technology (s.f.). Wallet. En *Computer Security Resource Center's Glossary*. Recuperado el 2 de febrero de 2022, de

<https://csrc.nist.gov/glossary/term/wallet>

Sandberg, J. (26 de noviembre de 2018). *Blockchain Fundamentals*

[Curso en línea]. Pluralsight.

<https://app.pluralsight.com/library/courses/blockchain-fundamentals>

Bibliografía

Accenture. (18 de agosto de 2020). *Blockchain in Review*

[Curso en línea]. Pluralsight.

<https://app.pluralsight.com/library/courses/tq-blockchain-questions>

Bitso. (23 de noviembre de 2021). *¿Cuáles son los límites/niveles para fondear mi cuenta o retirar?* Centro de Ayuda Bitso.

<https://help.bitso.com/es-LA/support/solutions/articles/1000161437--cu%C3%A1les-son-los-l%C3%ADmites-niveles-para-fondear-mi-cuenta-o-retirar->

Bitso. (24 de septiembre de 2021). *Bitso para cuentas empresariales.*

Centro de Ayuda Bitso.

<https://help.bitso.com/es-LA/support/solutions/articles/11000106815-bitso-para-cuentas-empresariales>

Buterin, V. (2013). *A Next-Generation Smart Contract and Decentralized Application Platform.* Ethereum.org.

Recuperado el 25 de enero de 2022 de

<https://ethereum.org/en/whitepaper/>

Canal Educablock. (20 de enero de 2021). *Aprende a cómo rastrear en Blockchain: Exploradores de bloques explicados en 5 minutos*

[Archivo de Video]. Youtube.

<https://youtu.be/2JkzHbDLyZg>

Canal Whiteboard Crypto. (26 de octubre de 2021). *What is Web 3.0?*

(Explained with Animations) [Archivo de Video]. Youtube.

<https://youtu.be/nHhAEkG1y2U>

Driscoll, S. (14 de junio de 2016). *Introduction to Bitcoin and Decentralized Technology* [Curso en línea]. Pluralsight.

<https://app.pluralsight.com/library/courses/bitcoin-decentralized-technology>

Ethereum. (s.f.). *Solidity docs, vo.8.7*. Soliditylang.org.

Recuperado el 8 de febrero de 2022 de

<https://docs.soliditylang.org/en/vo.8.7/#>

Haunts, S. (13 de febrero de 2018a). *Blockchain – Principles and Practices* [Curso en línea]. Pluralsight.

<https://app.pluralsight.com/library/courses/blockchain-principles-practices>

Hoffmann, T. y Watchulonis, M. (Directores). (2020). *Cryptopia: Bitcoin, Blockchains, and the Future of the Internet* [Documental]. 3DCH Media; Norddeutscher Rundfunk; Studio Hamburg Enterprises.

Mushketyk, I. (6 de diciembre de 2018). *Developing Applications on Ethereum Blockchain* [Curso en línea]. Pluralsight.

<https://app.pluralsight.com/library/courses/ethereum-blockchain-developing-applications>

Nakamoto, S. (31 de octubre de 2008). *Bitcoin: A Peer-to-Peer Electronic Cash System* [Archivo PDF]. Bitcoin Project.

<https://bitcoin.org/bitcoin.pdf>

Ravi, J. (16 de noviembre de 2018). *Deploying Ethereum with AWS Blockchain Templates* [Curso en línea]. Pluralsight.

<https://app.pluralsight.com/library/courses/aws-blockchain-ethereum-deploying-templates>

Apéndice A - Acrónimos

2FA: Two Factor Authentication

ATM: Automated Teller Machine

BTC: Bitcoin

CAPTCHA: Completely Automated Public Turing test to tell Computers and Humans Apart

CLABE: Clave Bancaria Estandarizada

DAPP: Distributed Application

EOA: Externally Owned Account

ETH: Ether

EVM: Ethereum Virtual Machine

FinTech: Financial Technology

IDE: Integrated Development Environment

IoT: Internet of Things

MiPyMes: Micro, Pequeñas y Medianas Empresas

NIP: Número de Identificación Personal

NSA: National Security Agency

PoS: Proof of Stake

PoW: Proof of Work

QR: Quick Response

SHA: Secure Hash Algorithms

SPEI: Sistema de Pagos Electrónicos Interbancarios

XRP: Ripple

Apéndice B - Glosario de Términos

Blockchain:

Un libro contable digital distribuido de transacciones firmadas criptográficamente que se agrupan en bloques. Cada bloque está criptográficamente vinculado al anterior después de la validación y se somete a una decisión consensuada. A medida que se agregan nuevos bloques, los bloques más antiguos se vuelven más difíciles de modificar, creando resistencia a la manipulación. Los nuevos bloques se replican en copias del libro contable dentro de la red, y cualquier conflicto se resuelve automáticamente utilizando reglas establecidas (National Institute of Standards and Technology [NIST], s.f., definición 1).

Criptomoneda:

Un activo/crédito/unidad digital dentro del sistema, que se envía criptográficamente de un usuario de la red Blockchain hacia otro. En el caso de la creación de criptomonedas (recompensa por minar), el nodo de publicación incluye una transacción que envía la criptomoneda recién creada a uno o más usuarios de la red Blockchain. Estos activos se transfieren de un usuario a otro mediante el uso de firmas digitales con pares de claves asimétricas (NIST, s.f., definición 2).

Exchange:

Un exchange de criptomonedas es la plataforma o punto de encuentro donde se realizan los intercambios de estas a cambio de dinero fíat o de otras criptomonedas. En estas casas de cambio en línea es donde se genera el precio de mercado que marca el valor de las criptomonedas en base a la oferta y demanda (Bit2Me, s.f.).

Hash:

Una función que mapea cadenas de bits a cadenas de bits de longitud fija, satisfaciendo las siguientes dos propiedades: es computacionalmente inviable encontrar para una salida dada, una entrada que se mapee a esta salida; y es computacionalmente inviable encontrar para una entrada dada, una segunda entrada que corresponda a la misma salida (NIST, s.f., definición 3).

Smart Contract:

Una colección de código y datos que se implementa mediante transacciones firmadas criptográficamente en la red Blockchain. El Smart Contract es ejecutado por nodos dentro de la red Blockchain; todos los nodos deben obtener los mismos resultados para la ejecución, y los resultados de la ejecución se registran en el Blockchain (NIST, s.f., definición 4).

Wallet:

Una aplicación utilizada para generar, gestionar, almacenar o utilizar claves públicas y privadas. Una Wallet se puede implementar como un módulo de Software o Hardware (NIST, s.f., definición 5).

Apéndice C - Índice de figuras

Figura 3-1: Cajeros bitcoin en México	16
Figura 3-2: Ejemplo de código QR y dirección bitcoin	19
Figura 3-3: Chivo Wallet	20
Figura 3-4: Generación de Wallet de papel	21
Figura 3-5: Generación de Brain Wallet	22
Figura 3-6: Hardware especializado para minado de criptomonedas	23
Figura 3-7: Explorador de Blockchains: Blockchair.com	25
Figura 4-1: Arquitectura de aplicaciones tradicionales y descentralizadas	30
Figura 5-1: Bitso: página principal	36
Figura 5-2: Bitso: formulario para crear una cuenta	37
Figura 5-3: Bitso: verificación de correo electrónico	38
Figura 5-4: Bitso: validación de datos	39
Figura 5-5: Bitso: verificación de teléfono	40
Figura 5-6: Bitso: página de límites de cuenta	41
Figura 5-7: Bitso: Wallet	42
Figura 5-8: Bitso: seguridad de cuenta	43
Figura 5-9: Bitso: venta automática de bitcoin	45
Figura 5-10: Sitio para generar códigos QR bitcoin	46
Figura 5-11: Bitso: Wallet de bitcoin (BTC)	47
Figura 5-12: Bitso: opciones para depositar bitcoin	48
Figura 5-13: Bitso: dirección bitcoin (BTC) y código QR para depósito	49
Figura 5-14: Envío de bitcoin: definición de monto y selección de red	50
Aspectos generales sobre la adopción de la tecnología Blockchain en MiPyMes mexicanas	95

Figura 5-15: Envío de bitcoin: lectura de código QR y verificación de dirección	51
Figura 5-16: Envío de bitcoin: confirmación de transferencia y verificación de identidad	52
Figura 5-17: Envío de bitcoin: confirmación y notificación de operación	53
Figura 5-18: Recepción de bitcoin: notificación de operación	54
Figura 5-19: Recepción de bitcoin: nuevo balance en Wallet	55
Figura 5-20: Bitso: ejemplo de código QR y dirección ether (ETH)	56
Figura 5-21: Bitso: ejemplo de código QR, destination tag y dirección ripple (XRP)	57
Figura 5-22: MetaMask: instalación de extensión en Google Chrome	58
Figura 5-23: MetaMask: pantalla de bienvenida	59
Figura 5-24: MetaMask: importación o creación de Wallet	60
Figura 5-25: MetaMask: definición de contraseña	61
Figura 5-26: MetaMask: generación de frase secreta de recuperación	62
Figura 5-27: MetaMask: confirmación de frase secreta de recuperación	63
Figura 5-28: MetaMask: balance de Wallet en la red principal	64
Figura 5-29: MetaMask: selección de red de prueba Ropsten	65
Figura 5-30: Obtención de test ether para red de prueba Ropsten y balance de Wallet	66
Figura 5-31: Remix IDE: explorador de archivos	67
Figura 5-32: Remix IDE: compilador Solidity	68
Figura 5-33: Remix IDE: despliegue y ejecución de transacciones	69
Figura 5-34: Compilación del contrato EventoAdopcion.sol	72
Aspectos generales sobre la adopción de la tecnología Blockchain en MiPyMes mexicanas	96

Figura 5-35: Conexión del Remix IDE con MetaMask	74
Figura 5-36: Opciones de configuración para despliegue del Smart Contract	75
Figura 5-37: Gas estimado para despliegue del Smart Contract	75
Figura 5-38: Confirmación de despliegue del Smart Contract e interfaz para su utilización	76
Figura 5-39: Interacciones de tipo call	77
Figura 5-40: Interacción de tipo transact: función cambiaDuenoMascota	78
Figura 5-41: Transacción satisfactoria: cambiaDuenoMascota	79
Figura 5-42: Advertencia de fallo de transacción cambiaNombreMascota desde IDE	80
Figura 5-43: Advertencia de fallo de transacción esterilizaMascota desde IDE	81
Figura 5-44: Confirmación de transacción esterilizaMascota	81
Figura 5-45: Transacción fallida: esterilizaMascota	82
Figura 5-46: Transacción fallida: esterilizaMascota en Etherscan	82
Figura 5-47: Historial de transacciones realizadas en el Smart Contract	83

Apéndice D - Otros recursos

- Código fuente del Smart Contract:

<https://ropsten.etherscan.io/address/0x2f745f5ecf9860091e65206a14106f88b4c18970#code>

Si consideras que la información presentada en esta investigación te fue útil y deseas poner en práctica lo aprendido, considera utilizar los siguientes dos enlaces, los cuales tienen por objetivo recolectar fondos destinados a causas de bienestar animal en mi localidad:

- Enlace de programa de referidos para creación de cuenta en Bitso:

<https://bitso.com/register?ref=tvkd>

- Código QR y dirección bitcoin (BTC) para donativos:

