UNIVERSIDAD DON BOSCO

VICERRECTORÍA ACADÉMICA

FACULTAD DE INGENIERÍA

TRABAJO DE GRADUACIÓN PARA OPTAR AL GRADO DE

Maestro(a) en Seguridad y Gestión de Riesgos Informáticos

ANTEPROYECTO

Modelo de Comercio electrónico C2C con registro de transacciones sobre tecnología Blockchain

PRESENTADO POR

Dimas López, Hugo Ernesto

Flores Figueroa, Nohemy Nathaly

Flores Romualdo, Néstor Gerardo

ASESOR

Rudiger Fogelbach

Antiguo Cuscatlán, La Libertad, El Salvador, Centro América

Agosto 2021

Índice de contenido

Índice de tablas

Índice de imágenes

1. Fundamentos de comercio electrónico

1.1. Conceptos y características

El *e-commerce* o “comercio por internet, comercio en línea” consiste, básicamente, en una transacción de compra y venta de algún servicio o producto, a través de internet, que deriva de las necesidades, deseos y demandas del cliente, ofertas de mercado, productos, servicios y experiencias.

El *e-commerce*, esta expresión ha existido desde que las tecnologías de la información se fusionaron con las actividades económicas actuales. Si bien es cierto, el término es nuevo en la industria, las transacciones electrónicas más comunes, como envío o retiro de dinero por medio del sistema financiero o un cajero automático, forman parte de las actividades del comercio electrónico.

A este amplio esquema, se han adherido herramientas diversas que facilitan y añaden ventajas como versatilidad de comunicación, facilidad de transacción o negociación.

Por otro lado, el *e-commerce* agrega valor en áreas como la publicidad de servicios o productos, una herramienta que forma parte, hoy en día, de las estrategias empresariales más importantes para las ventas y por ende, para el flujo de dinero dentro de la Compañía.

El *e-commerce*, para algunos autores, lo definen de la siguiente manera:

“La disponibilidad de una visión empresarial apoyada por la avanzada tecnología de información para mejorar la eficiencia y la eficacia dentro del proceso comercial” [1]

“El comercio electrónico se basa en tecnologías como el comercio móvil, la transferencia electrónica de fondos, la gestión de la cadena de suministro, la comercialización por Internet, el procesamiento de transacciones en línea, el intercambio electrónico de datos, los sistemas de gestión de inventarios y los sistemas automatizados de reunión de datos.” [2]

Características del comercio electrónico

Las características principales que podemos destacar del *e-commerce* son:

- Las transacciones se realizan de forma remota
- Mínimo manejo de documentación física.
- Automatización de transacciones.
- El personal de las empresas no interviene físicamente en la mayoría de transacciones.

Actualmente la manera de comerciar se caracteriza por la mejora constante en los procesos de abastecimiento y, como respuesta a ello, los negocios están cambiando mundialmente, así como en la organización y sus operaciones. Lo que se transforma en una nueva economía que se caracteriza porque, no existen límites geográficos y fomenta una cultura de autoservicio.

Además, esto facilita las tareas siguientes:

- Sitios web de compras en línea para ventas minoristas directas a los consumidores.
- Proporcionar o participar en mercados en línea, que procesan negocios de terceros para ventas de consumidor a consumidor.
- Compra y venta de empresa a empresa.

Las características antes mencionadas forman parte fundamental de lo que ahora se conoce como “comercio electrónico”. Por otro lado, se puede estudiar este modelo de comercio con base a otro enfoque, y este proviene del llamado modelo 5-C (Zwass 2014). [3]

Define el comercio electrónico por cinco dominios de actividad cuyas denominaciones inician con la letra "C", y estas son:

Comercio: Con el amplio movimiento hacia los sistemas empresariales habilitados en la web, ahora existe una cadena de suministros universal, clientes y proveedores establecen ahora los términos de la transacción y facilitan las mismas.

Colaboración: El Internet es un vasto nexo o red de relaciones entre empresas e individuos, en este se crean o surgen colaboración, trabajos entre otros, sin limitantes de tiempo y frontera.

Comunicación: El comercio móvil es de rápido crecimiento, permite la conectividad en contexto como publicidad y productos sensibles a la ubicación, en el ámbito de las comunicaciones, este sirve como canal de distribución para productos digitales.

Conexión: Gracias a los avances tecnológicos las plataformas de desarrollo de software comunes, muchas son de dominio de código abierto, esto permite que un amplio espectro de empresas aproveche los beneficios de los softwares desarrollados.

Computación: La infraestructura de Internet permite el intercambio a gran escala de recursos informáticos y de almacenamiento, conduciendo así a la implementación de la idea de la informática de utilidad.

1.2. Antecedentes del comercio electrónico

El uso de redes para intercambiar dinero y transferencias comenzó a finales de la década de 1950 con el desarrollo de transferencias electrónicas de fondos (EFT, por sus siglas en inglés *Electronic Fund-Transfer*) eran la transmisión electrónica de cuentas o información a través de redes de comunicación privadas. [1]

El intercambio electrónico de datos (EDI, por sus siglas en inglés *Electronic Data Interchange*), mediante el cual empresas e individuos intercambian información legible por computadora, en un formato estándar para otras empresas, fue la forma más temprana de comercio electrónico. A finales de la década de 1960, el intercambio electrónico de datos se utilizó para reducir la cantidad de tiempo y esfuerzo para ingresar datos como facturas, órdenes de compra y otros documentos. [1]

Dado que este tipo de información a menudo tiene un formato regular, los sistemas fueron diseñados para leer estos documentos electrónicamente. Los formatos tuvieron que ser

acordados, y para muchas industrias, como el transporte aéreo y marítimo, que son de naturaleza global, este enfoque unificado fue importante.

Las empresas que participan en EDI, se denominan socios comerciales. Los mayores usuarios del comercio electrónico eran tradicionalmente agencias gubernamentales y grandes corporaciones, debido al alto costo de implementación. No fue hasta finales de la década de 1990, que EDI significaba la compra de costosos programas informáticos y hardware o establecimiento de conexiones de red directas con todos los socios comerciales.

A pesar de que algunas empresas ofrecieron redes de valor agregado (VAN, por sus siglas en inglés *Value Add Network*) como sistemas para realizar EDI, suscribiéndote a tales, las *VAN* tenían un costo elevado.

Entre los años 1997 y 2000 se iniciaron más de 12.000 negocios relacionados con Internet. Muchas de estas empresas quebraron, debido a que no tenían modelos de ingresos suficientemente sólidos para generar suficientes ingresos para mantener su negocio. [1]

A medida que más y más empresas compiten por un número fijo de bienes o ideas, los negocios de Internet se sobrevaloran y también se implementaron muchas malas ideas. Para el año 2000, el negocio de Internet había comenzado a experimentar una recesión. Miles de empresas quebraron por falta de ingresos por publicidad, esto significaba que no podían mantener su promesa inicial.

La evolución del *e-commerce* ha sido estudiada y seguida por varios investigadores y partes interesadas en el campo. Dados los avances en Tecnología de la Información, más precisamente con el desarrollo de Internet, a partir de la década de 1990, se percibió un amplio abanico de posibilidades, con un marcado énfasis en la comunicación.

Sin embargo, (Galinari et al., 2015) [3] defienden que el comercio electrónico tiene su primera fase, en la década de 1970, cuando el comercio electrónico se restringía a las operaciones entre grandes corporaciones que establecen entre sí redes de comunicación privadas y, mediante sistemas electrónicos de transferencia de fondos, que realizaban transacciones financieras e intercambios de documentos por vía electrónica.

El nacimiento del *e-commerce* se puede dividir en cuatro fases:

Fase uno, las organizaciones utilizaron las funcionalidades de Internet para procesos de divulgación de información sobre sus productos y servicios. Ese fue el estímulo inicial para el desarrollo de *e-commerce.* [4]

Según el autor, la **fase dos** fue recibir pedidos y enviar información e instrucciones sobre la utilización de sus productos y servicios. En esta fase, la logística provocó su primer impacto en las empresas.

La **fase tres** de la evolución, según (Albertin, 2012) [4] fue la distribución de productos y servicios mediante el uso de Tecnología de la Información (TI). En esta fase, algunos productos comenzaron a comercializarse digitalmente como, por ejemplo, música y software.

Para **la última fase**, llega la fase que consolida al *e-commerce*, con la interacción entre el vendedor y el consumidor, no solo transmitir datos o entregar productos y servicios únicamente. [5]

Con el avance de la tecnología de la información y el uso generalizado de Internet, dicha interacción permitió que el simple usuario de Internet se convirtiera en un consumidor potencial, dadas las posibilidades del *e-commerce*.

Esta herramienta permitió una verdadera revolución en la forma de comercializar productos, servicios e información, aportando más comodidad y gran variedad de ofertas y opciones para el consumidor, pero también para el vendedor que se inserta en esa práctica de mercado, pues es normal que las empresas sufran transformaciones. en su estructura, y la globalización contribuyó a esa fuerte tendencia.

El aumento de la competitividad, la necesidad de producir innovación y la creciente demanda de los consumidores -todos ellos traídos por la globalización- han culminado con la aparición de formas más modernas de gestión empresarial.

Según autores, el avance en el acceso a Internet de banda ancha en los últimos años se ha vuelto importante al desarrollo del *e-commerce*. La llega de la tecnología 3G y 4G, a partir de 2012 en Latinoamérica, especialmente en Brasil, dio acceso a Internet de alta velocidad a través de dispositivos móviles, como teléfonos inteligentes y tabletas, tecnología que permite

al consumidor con mayor facilidad realizar búsquedas de precios para varios productos o servicios, dependiendo su necesidad. [6]

El desarrollo del *e-commerce* sigue las etapas de evolución del entorno digital, evolución que debe entenderse y garantizarse a través de los aspectos que deben tenerse en cuenta en el uso del *e-commerce*, con el objetivo de asegurar el uso de sus aportes.

Por lo tanto, el desarrollo de la tecnología de la información se ve afectado por la evolución del *e-commerce.* Otra dimensión que completa el análisis sobre las fases del *e-commerce* es su aplicación en los procesos de negocio (Albertin, 2000). [6] [5]

De esta manera, el comercio electrónico se convirtió, más que una tendencia, en una realidad. Las empresas modernas buscan cambiar su estructura para cumplir con las demandas de los nuevos consumidores, una vez que estos valoran cada vez más la comodidad. Para tal fin, el uso de nuevas tecnologías es una estrategia muy común.

1.3. Ventajas y desventajas del comercio electrónico

Ventajas

Cliente	**Proveedor**
• Horario comercial flexible • No hay colas de espera (si la red está disponible y el software está adecuadamente diseñado) • Comprar en casa (el cliente no necesita salir de su hogar) • Ofertas globales	• Comunicación rápida con el cliente • Potenciales clientes a través de visibilidad global • No hay intermediarios (y altas comisiones)

Desventajas	
Cliente	**Proveedor**
• Riesgos de seguridad. • Robo de datos (p. ej., robo de cuenta o números de tarjetas de crédito) • Robo de identidad (actuando bajo el nombre o identidad del usuario) • Robo de identidad • Fraude (p. ej., pedido confirmado, factura tiene que ser pagada, pero los bienes nunca son entregados)	• Mayor costo logístico (los bienes deben enviarse a la ubicación del cliente) • Anonimato de los clientes (el proveedor pierde la oportunidad de crear anuncios dirigidos)

Tabla 1: Ventajas y desventajas del comercio electrónico.

1.4. Modelos de negocio de comercio electrónico

Como sabemos la infraestructura del Internet alteró la economía del mundo entero, eliminando las barreras socio-culturales y proporcionó una vasta y nueva red que ofrece un increíble poder de conectividad e inmediatez surgiendo de esta forma el Comercio Electrónico.

El comercio electrónico a través de Internet puede ser complementario de los negocios tradicionales o representar una línea completamente nueva [7]; gracias a esta transformación digital, ahora resulta indispensable que el comercio sea innovador y creativo, que replantee y/o reinvente los modelos de negocio tradicionales.

Básicamente un modelo de negocios es una herramienta que sirve para representar las actuaciones y misiones de una empresa y definir el método de hacer negocios por el cual una empresa puede generar ingresos y sostenerse.

Si bien no existe una definición 100% consensuada en el mundo académico sobre que es un modelo de negocio, al menos sí que hay un acuerdo general sobre los elementos principales que deben constituir el modelo.

"Un modelo es una representación simplificada (en muchas ocasiones de forma gráfica) de la lógica del negocio, es decir, la descripción de la lógica en que cada negocio ofrece sus productos o servicios a los clientes, de cómo llega a estos, de su relación con ellos y por supuesto de cómo gana dinero". [8]

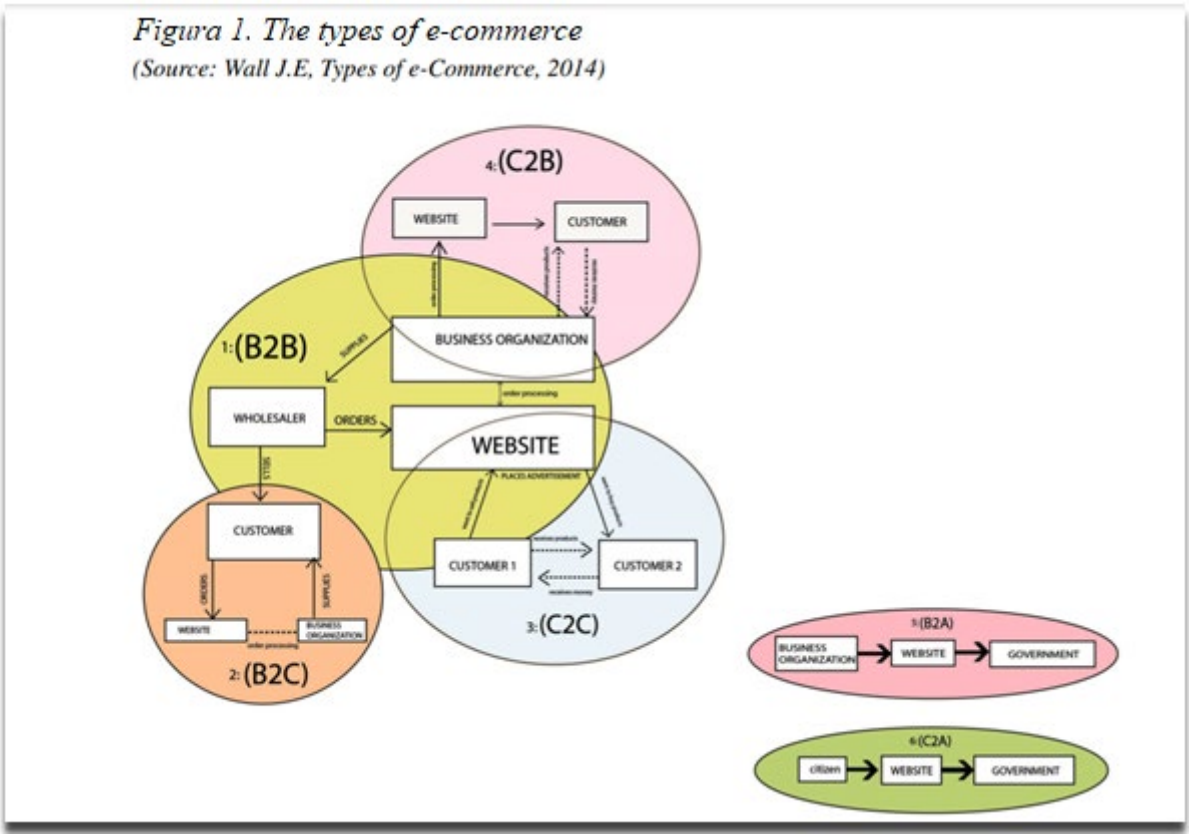
Se cree en gran medida que los modelos comerciales determinan el éxito de una empresa electrónica.

Para un modelo de negocio *e-business*, la sostenibilidad, la innovación y flexibilidad, deben ser herramientas imprescindibles. Abiertos a cambios de mentalidad que impliquen reconsiderar la relación con los clientes, la cadena de suministro, las alianzas y la redefinición de productos, servicios y procesos desde una nueva perspectiva.

Modalidades del Comercio Electrónico

Se puede dividir el comercio electrónico en seis grandes grupos según sus funciones y el tipo de agentes que intervienen en la relación de intercambio.[9]

Imagen 1: The types of e-commerce



***Fuente* - https://www.researchgate.net/publication/335821863**

Como se puede observar en el diagrama, *Imagen 1*, se plantean los diferentes modelos y procesos de comercio electrónico y cómo se vinculan entre sí a través de conexiones mutuas, usando como medio común la web. [9]

- B2B (*Business-to-Business*): Empresas que hacen negocios entre ellas. El *business-to-business* aplica a la relación entre un fabricante y el distribuidor de un producto y también a la relación entre el distribuidor y el comercio minorista. [10] Las empresas más conocidas con este modelo son Alibaba, *Staples, Quill, Office max, Medline*.
- B2C (*Business-to-Consumer*): Empresas que venden al público en general, a los consumidores como personas naturales. [10] En comparación con los métodos de venta minorista tradicionales, B2C es mucho más conveniente, ya que proporciona a los clientes la información necesaria del producto sin salir de casa.

  La mayoría de los clientes también encuentran productos mucho más baratos en comparación con otros métodos de compra con entrega y procesamiento rápidos de sus pedidos y un mejor servicio al cliente. La empresa Amazon es la más reconocida con este modelo. [9]
- C2C (*Costumer to Costumer*): Este tipo de comercio electrónico implica las transacciones de una variedad de bienes y servicios entre un consumidor y otro. Las personas utilizan una plataforma electrónica que actúa como un tercero, para publicitar sus productos o para comprar los productos a otros consumidores.

  Existe una gran cantidad de plataformas electrónicas que se dedican únicamente a realizar las transacciones C2C, algunos ejemplos incluyen expatriates.com, *E-bay* y Encuentra24. [9]

- C2B (*Consumer-to-Business*): Es el modelo de negocio de comercio electrónico en el que los consumidores pueden ofrecer productos y servicios a las empresas, y las empresas pagan a los consumidores. Este modelo de negocio es la inversa del modelo de negocio tradicional B2C. [10]

  Hay una serie de sitios web donde los diseñadores gráficos presentan algunas muestras de logotipos de empresas y las empresas eligen el mejor entre ellos para adquirirlo. *iStockPhoto* es un claro ejemplo de C2B, conocido por vender imágenes, fotografías con derechos de autor, medios y elementos de diseño (Wall, 2014) [9]

- B2G (*Business-to-Government*): Empresas que venden u ofrecen sus servicios a las instituciones del gobierno. Los ayuntamientos, diputaciones y otras instituciones oficiales, pueden contactar con sus proveedores, comparando productos y realizando pedidos por medio de un proceso simple y estandarizado. [10]
- C2G *(Consumer-to- Government*): El comercio electrónico C2G implica todo tipo de transacciones en línea entre los consumidores y la administración pública. Se utiliza principalmente para el pago de impuestos, educación a distancia, trámites y fijación de citas o recolección de información de los centros de salud. Esta es también una facilidad proporcionada por el gobierno para el apoyo y la comodidad de los ciudadanos.[9]

Tal como se define en el título de la investigación, nuestra propuesta se desarrolla para el Modelo de Comercio Electrónico C2C, por lo que profundizaremos más sobre este modelo.

1.5. Modelo de negocio Customer to Customer (C2C)

Definición y Características

El modelo de negocio C2C (*Costumer to Costumer*), permite el intercambio de bienes y servicios entre consumidores, existiendo entre ellos una relación horizontal. Esta relación de negocios entre los consumidores es muy espontánea y natural.[11]

En este tipo de comercio electrónico, los consumidores finales promueven sus productos y servicios en un sitio web o aplicación, donde otros consumidores pueden acceder y realizar un intercambio estableciendo las condiciones entre ellos mismos.[12]

El sitio web es solo un lugar de intercambio, solo están ahí para coincidir consumidores. No tienen que comprobar la calidad de los productos que se ofrecen y obtienen una comisión por cada venta efectiva realizada. Las formas en las que se produce esa compraventa pueden ser mediante subasta, intercambio, entre otros.[13]

Ventajas para los compradores. [14]

- Alta disponibilidad: Los sitios web C2C están disponibles 24/7, los usuarios pueden hacer compra / venta en cualquier momento y lugar, solo se requiere conexión a internet.

- Mayor rentabilidad: La principal ventaja de este modelo de *e-commerce* es la posibilidad de realizar transacciones de bajo costo, donde ambas partes pueden sacar mucho provecho al no contar con intermediarios que elevan los precios. [15]
- Relación directa: Las formas de pago dependen del convenio entre ambas partes y, por lo general, la entrega se hace en el mismo momento del pago, eliminando los tiempos de espera que suelen presentarse en el comercio B2C (*Business to Consumer*). [15]
- Valor Agregado: Los intermediarios, cuando existen, se ven obligados a crear valor agregado en sus productos o servicios.
- Rapidez: sobre todo si es factible en el momento de descargar de la red el producto digital.
- Sistema de valoraciones: Base de datos que permiten a los compradores evaluar la experiencia de compra, atención del vendedor y calidad del producto. Con el fin de servir de guía a futuros compradores. Es un sistema potencial, no todas plataformas lo incluyen necesariamente.

Ventajas para los vendedores. [14]

- Negociación directa con posibles compradores.
- Subastas.
- Venta de productos y/o servicios.
- Presencia global, es decir, posibilidad de vender en lugares remotos.
- Costos mínimos o nulos en algunos casos (ya que no se eroga en un canon de arrendamiento).
- Posibilidad de vender sin necesidad de estar completamente establecidos.
- Creación de un historial de referencias para futuros tratos, intercambios o compra-ventas.

Desventajas del C2C [14]

- Peligro de fraude: En el entorno de comercio electrónico C2C es posible y de hecho bastante común, tratar con vendedores anónimos, cuya confiabilidad se desconoce.
- Mala calidad: El intermediario no se responsabiliza ni verifica que los productos ofertados sean de buena calidad o que estén en buen funcionamiento.
- Problemas de entrega: Daños en el envío por parte de la paquetería. (Dependiendo de la empresa que ofrece el servicio de Courier y es inherente a cualquier entrega).

- Cobros inesperados: Gastos de envío a veces excesivos dependiendo del producto y paquetería. (Dependiendo de la empresa que ofrece el servicio de Courier y es inherente a cualquier entrega).
- Operaciones ilegales: Compra-venta de productos o servicios de dudosa procedencia.
- Desconfianza: Los consumidores son reacios a comprar a vendedores web desconocidos por temor a ser víctimas de cargos fraudulentos, productos falsos o de baja calidad, dificultades para devolver productos defectuosos o erróneos, etc.)

1.6. Desafíos actuales del comercio electrónico.

Con el propósito de asegurar la continuidad y extensión del comercio electrónico, se enumeran algunos desafíos técnicos, administrativos, legales y de negocio que el *e-commerce* debe afrontar y superar para no frenar su crecimiento.

Protección al consumidor.

Uno de los principales desafíos del *e-commerce* es generar confianza en el uso de las plataformas y tiendas virtuales, especialmente en los pagos electrónicos con tarjetas de crédito o procesadores de pago (como *Paypal* o *2checkout* entre otros). Esta confianza implica que se adopten medidas tecnológicas y especialmente marcos jurídicos y regulatorios que respalden al consumidor ante transacciones electrónicas fraudulentas y brindando mayor seguridad y soporte.

El consumidor debe exigir, como derechos mínimos, que se le proporcione información clara y actualizada del producto que se desea adquirir, de la empresa que lo ofrece, de los gastos detallados a los que se incurre al adquirir el producto, de un mecanismo de rastreo y seguimiento del producto durante y posterior a la compra, a exigir la calidad del producto de acuerdo a la descripción, derecho al registro y seguimiento de reclamos, entre muchos otros.

Protección y privacidad de los datos.

Este desafío está relacionado a la manera en que las plataformas tecnológicas destinadas al *e-commerce* manejan la información que los actores suministran, considerando que están expuestas a riesgos y amenazas a la integridad y confidencialidad, especialmente cuando

esta información contempla datos sensibles de los usuarios como correos electrónicos, nombres de usuario, números de cuenta, números de tarjeta de crédito o débito, entre otros.

Dado que existen casos en los que la información de los usuarios ha sido manipulada sin consentimiento de estos y con fines de influencia (como el caso de Cambridge Analítica y su influencia en las elecciones presidenciales de los Estados Unidos en el año 2015 [16]), la protección y privacidad de la información se vuelve un factor clave en el uso de las plataformas de *e-commerce*.

Es necesario que las plataformas y tiendas virtuales generen en sus usuarios un nivel de confianza adecuado para operar dentro de estas. Para ello, es importante que las plataformas expongan de forma clara las técnicas utilizadas para recopilación, almacenamiento, tratamiento, eliminación, consentimiento expreso de los actores y de si la información suministrada es compartida con terceros.

A nivel jurídico y regulatorio para el control y privacidad de la información, el comercio electrónico se ve reglamentado por La Regulación General de Protección de Información (GDPR por sus siglas en inglés *General Data Protection Regulation*) con vigencia a partir del 25 de mayo de 2018 para los países de la Unión Europea (UE) y para las organizaciones públicas y privadas que manejen información de personas residentes de la UE.

Seguridad en pagos dentro del e-commerce.

Una de las principales preocupaciones de los usuarios al realizar transacciones comerciales en línea, son aquellas relacionadas a los registros de pagos (que concluyan de forma exitosa), reversiones de pago, cargo de pagos dobles o no autorizados. Desde una perspectiva técnica, que las operaciones se realicen sobre redes y plataformas seguras que validen identidad, integridad y no repudio.

Ciberseguridad.

La seguridad en el *e-commerce* es sólo uno de los aspectos dentro de la seguridad de las TIC (Tecnologías de la Información y las Comunicaciones). Comprende el uso de componentes que garanticen la protección de las computadoras, sistemas, datos y redes de

comunicación ante terceros no autorizados y mal intencionados como los cibercriminales, tanto de los que utiliza el proveedor como el consumidor.

Los desafíos que el *e-commerce* debe gestionar son los relacionados a la seguridad de los servidores donde se alojan las plataformas y tiendas de *e-commerce*, las cuales incluyen la red interna y periférica para disponer del servicio, usos de certificados de seguridad entre otras técnicas recomendadas.

Por otro lado, gestionar las amenazas a las que se encuentra expuesto el negocio a través de sus clientes potenciales, tales como ataques de *Phishing*, ataques de denegación de servicio (DoS por sus siglas en inglés *Denial of Service*) o denegación de servicio distribuido (DDoS por sus siglas en inglés *Distributed Denial of Service*), Fraudes con tarjetas de crédito, entre otros.

Si bien es cierto, no es posible garantizar la seguridad en su totalidad, si es posible definir procesos de gestión y tratamiento del riesgo y materialización de vulnerabilidades y amenazas. De esta manera, se proyecta seguridad y confianza en el uso de las plataformas y tiendas virtuales.

Propiedad Intelectual.

El *e-commerce* permite desarrollar catálogos de productos mediante la implementación de tiendas virtuales basadas en sitios web y/o aplicaciones móviles, las cuales están dotadas de textos, contenidos multimedia y otros elementos tecnológicos que forman parte de la estrategia de venta del proveedor, haciendo llamativo adquirir los productos por parte de los clientes en potencia.

Dentro de estas plataformas donde se desarrolla el *e-commerce*, cobra vital importancia el tema de la propiedad intelectual, en relación con los elementos con los que el cliente interactúa y visualiza, tanto sobre el producto como para la promoción de este.
El concepto de propiedad intelectual, según la OMPI (Organización Mundial de la Propiedad Intelectual) es:

“La propiedad intelectual se relaciona con las creaciones de la mente: son las invenciones, obras literarias y artísticas, así como símbolos, nombres e imágenes utilizados en el comercio” [17]

Por lo anterior, es importante indicar que la propiedad intelectual es relevante en el *e-commerce*, dado que se genera una transferencia de bienes basados en propiedad intelectual y que los productos adquiridos están ligados a tipos de uso o licencias.

Podemos considerar como elementos a proteger el conjunto de imágenes, audio, videos, textos, marcas, iconografía, sistemas informáticos, redes de comunicación entre otros que forman parte del *e-commerce*.

El proteger la propiedad intelectual comprende un desafío trascendente para la continuidad de este tipo de operaciones comerciales, de tal manera que existan leyes que respalden dicha propiedad y autoría, garantizando y persiguiendo delitos de falsificación y piratería.

Legislación.

Uno de los principales desafíos a los que se enfrenta el *e-commerce* es el relacionado al marco jurídico sobre el cual se sustentan las distintas transacciones de comercio sobre redes de comunicación tecnológicas o equivalentes, para cualquiera de los participantes dentro de esta relación comercial, independiente del rol que cumplan dentro de la operación (proveedor o consumidor).

Por un lado, los proveedores deberían estar obligados a proporcionar información clara, comprensible e inequívoca de los productos que se ofrecen y que les permitan a los consumidores tomar la decisión de entablar una relación de intercambio comercial, tal como se expresó en los apartados de protección al consumidor y protección y privacidad de datos. Por otra parte, los consumidores deben estar obligados a cumplir con los pagos, retribuciones y condiciones que el proveedor indique, para que la oferta del producto o servicio sea celebrada por mutuo acuerdo bajo la vía tecnológica.

Capital humano

La implementación y establecimiento de un negocio en línea requiere de personas capacitadas en distintas áreas, que garanticen la continuidad del negocio y un adecuado manejo de las operaciones diarias.

El capital humano con el que se debe contar incluye a expertos en tecnologías de la información que permitan implementar, establecer y mantener las plataformas tecnológicas tanto en continuidad, facilidad de uso y seguridad.

De igual manera, se debe contar con expertos en materia legal que permitan asesorar sobre las políticas y condiciones de uso de las plataformas y de los productos y que estén alerta ante los cambios jurídicos y regulatorios que puedan impactar la operación.

Por otro lado, se debe contar con expertos en comunicación y promoción de los productos en distintos medios, que permitan identificar y ejecutar campañas, así como analizar datos para identificación de tendencias.

Contar con el capital humano adecuado se vuelve crucial para la operación y continuidad del negocio en línea, dado que se juega con la reputación y veracidad tanto de la marca como del producto.

Consumidor internacional.

El alcance global que tiene el *e-commerce* implica que las tiendas virtuales trascienden de los usos y nichos locales a nivel país, por lo que se deben implementar cambios en las plataformas que permitan gestionar múltiples idiomas, aceptación de distintas formas de pago y monedas. De igual manera, se debe entender la necesidad de los segmentos de clientes de acuerdo a su región geográfica, brindando información adecuada y acorde a lo que el potencial cliente busca.

Las plataformas deben manejar y atender los cambios en las legislaciones de los distintos países y zonas, de tal manera que no se generen infracciones o incumplimientos normativos, incluso si el proveedor del producto no se encuentre en el mismo país o región geográfica de sus clientes potenciales.

Logística

A pesar de que el *e-commerce* permite realizar transacciones en línea, algunos de los productos que se ofrecen utilizan los mismos medios de distribución que el comercio

tradicional, por lo que se vuelve un desafío controlar las cadenas de suministro y transporte, los cuales pueden significar un costo alto en la operación.

Es posible que los proveedores, instituciones de gobierno y aduanas no cuenten con mecanismos de control automatizados por lo que vuelve engorroso los procesos de compra de materiales o envío de productos y por consecuencia aumentan los tiempos de espera.

Dado que la logística no es un proceso que se pueda controlar en su totalidad, es necesario que se consideren estos costos operativos y de tiempo en la oferta de los productos en las distintas plataformas.

Cuestiones sociales, culturales y políticas.

Las personas siguen teniendo desconfianza en el uso de plataformas electrónicas, no únicamente para la compra de productos en línea, sino también para procedimientos gubernamentales y legales. Adicional, podemos mencionar el bajo porcentaje de acceso de personas con acceso a internet [18], por lo que existe un segmento de mercado inaccesible por el *e-commerce*.

Se vuelve primordial poder educar a la población en los usos de las plataformas y los beneficios que representa este tipo de transacciones comerciales, y sobre todo generar la confianza necesaria para que realicen sus primeras compras y adopten estos medios para satisfacer sus necesidades.

1.7. Tendencias del comercio electrónico

Omnicanalidad.

La innovación en los procesos de venta con enfoque al cliente requiere una afinación continua de las formas de comercio tradicional y online (en conjunto). Los clientes interactúan y utilizan diferentes canales de venta, siendo usual, por ejemplo, que un consumidor agregue productos a un carrito de compras a través de la web y luego haga efectiva la compra a través de una aplicación móvil, otorgando de cierto empoderamiento al consumidor.

Ante esta necesidad de generar valor cuyo foco se centra en el cliente, se vuelve necesaria la implementación de la omnicanalidad como la clave de éxito en la estrategia de gestión de los consumidores y que se encuentra estrechamente relacionada con la tecnología.

Entendemos como omnicanalidad a la estrategia y gestión de canales que tienen como objetivo la integración y alineación de todos los canales disponibles, con el fin de brindar a los clientes una experiencia de usuario homogénea a través de estos [19][20].

La decisión de transformar un comercio como omnicanal, requiere de un proceso de planificación, análisis y recursos, de tal manera que se garantice una experiencia de usuario donde le sea indiferente usar un medio u otro.

Esta experiencia al cliente debe contemplar la conveniencia en los procesos de compra, consistencia de la marca y de los productos en los diversos canales, la agilidad y relevancia necesaria para el consumidor y en el empoderamiento del cliente en la toma de decisiones.

Dentro de los retos para la implementación de una estrategia omnicanal que se deben valorar mencionamos la inversión en tecnología, cambio de mentalidad en la organización y la consistencia entre inventarios y precios online como en el comercio tradicional.

Tecnología *Blockchain*.

Blockchain es una tecnología que permite la inmutabilidad y la integridad de datos en los que se mantiene un registro de las transacciones realizadas en un sistema a través de varios nodos distribuidos que están vinculados en una red de igual a igual [21]

Esta tecnología consiste en la creación de transacciones en entornos descentralizados que se registran en elementos de bloques de información que no cambian en el tiempo y que son validados por todos los participantes de una red.

Esto permite, mediante el uso de métodos de criptografía y algoritmos de consenso, generar confianza entre los participantes de la red, auditoria y transparencia en las transacciones registradas, escalabilidad operativa de la red y la eliminación/reducción de intermediarios.

Las características y ventajas que proporciona la red *Blockchain*, se convierten en una tecnología adecuada para ser aplicada dentro de los entornos de comercio electrónico,

especialmente en aquellos en los que se necesita implementar una arquitectura de *e-commerce* que otorgue seguridad, validad y permanencia del negocio.

Logística

Dentro del *e-commerce*, el proceso de entrega de productos ha manifestado cambios que permiten una mejor experiencia del cliente en cuanto a métodos de despacho y tiempos de entrega, por lo que las empresas colocan una especial atención en este factor diferenciador e implementan novedosos procesos de entrega de "última milla", los cuáles son considerados como los procesos más retadores dentro del proceso de distribución.

Dentro de estos nuevos métodos de entrega podemos mencionar el "*Click and collect*", en donde el cliente realiza la compra mediante las plataformas de *e-commerce* dispuestas en internet y le permiten recoger el producto en las tiendas físicas.

Esto le habilita al cliente vivir una experiencia "*express on line*" pero con las ventajas de poder validar las características físicas del producto en las tiendas y quioscos, además de seleccionar fecha, hora y forma de entrega del producto. Adicionalmente, este proceso permite la venta de productos complementarios directamente en tiendas, lo que genera un ingreso adicional para las empresas.

Por otra parte, ya se realizan entregas de paquetes en distancias cortas mediante el uso de drones. Un dron es una nave de vuelo no tripulada y manejada a distancia. El uso de drones representa un mecanismo de entrega ágil y a bajo costo, no es afectado por el congestionamiento vehicular ni malas condiciones de las carreteras, permitiéndole llegar a sitios remotos.

No es posible realizar entrega de grandes paquetes y tampoco es posible cubrir rutas de entrega que representen un tiempo mayor a la vida de la batería utilizada por el aparato. No obstante, es una de las opciones que se investigan a futuro por constituir un excelente recurso para cubrir un mayor volumen de entregas de forma eficiente.

Impresión 3D

La impresión 3D consiste en crear diseños tridimensionales a partir de un diseño digital mediante el uso de impresoras especializadas. Una impresora 3D es una tecnología que usa la fabricación aditiva por capas, permitiendo imprimir o, en este caso, recrear un objeto tridimensional, con distintos materiales, cuya información se envía desde un dispositivo [22].

Aunque esta tecnología no sea muy accesible en varios países, representa una gran oportunidad para el *e-commerce* ya que permite una recreación personalizada del producto de acuerdo a la necesidad del cliente. De igual manera, se reduce el inventario físico, ya que los productos son generados a demanda según la necesidad, a la vez que se reducen costos de envío de "última milla". Adicionalmente, se realiza el comercio de ideas y diseños descargables en lugar de productos terminados.

Una de las desventajas de la impresión 3D será la propiciación de la piratería digital afectando los derechos de propiedad industrial, no obstante, representará una excelente oportunidad como tecnología disruptiva para el *e-commerce*, que les permitirá a las empresas de todo tipo ofrecer y generar productos ajustados al cliente.

Inteligencia Artificial (IA) y *Big Data*

El análisis de la información acerca del cliente y sus comportamientos de compra, se ha vuelto importante para que las empresas generen ofertas especializadas conforme a las características de los nichos de mercado objetivos, personalizando los procesos de *marketing* y venta. Dentro de las tendencias actuales para este tipo de análisis encontramos a la Inteligencia Artificial IA y el *Big Data*.

Sistemas que identifican imágenes, programas que logran el reconocimiento de voz, celulares que distinguen un rostro o un tono de voz representan situaciones que actualmente hacen parte de la vida cotidiana; todo ello es posible por la aplicación de algoritmos que procesan información similar a como lo haría el cerebro humano y a grandes volúmenes de información auxiliándose del *Big Data.*

Las tecnologías de la IA aplicadas al *marketing* evolucionan la forma de operar de las empresas, así como su manera de interactuar con los clientes al mejorar la capacidad de procesamiento y análisis de transacciones. Con la aplicación de estas tecnologías, se logra

la personalización de la venta y una sensación de relación interpersonal con el cliente, ofreciendo al cliente lo que realmente necesita.

Las tecnologías de la IA aplicadas al *e-commerce* generan ecosistemas enfocados en diseñar estrategias competitivas con facultades para predecir comportamientos y una experiencia más reconfortante para los clientes.

Realidad aumentada (RA)

Por medio de la Realidad Aumentada se logra crear un ambiente interactivo de comunicación que sobrepone objetos como una animación o un video a un escenario real y que, por ejemplo, a través de la proyección hecha con una cámara web y proyectado en dispositivo electrónico, se logra proporcionar información más llamativa, interactiva y amena a las personas [23].

La RA se ha venido utilizando hace tiempo principalmente en la publicidad y el *marketing*, con el auge del uso de los dispositivos móviles y el gran nivel de acceso a Internet, esta tecnología es aplicable a varios sectores de la economía como lo son el mercado editorial, vestuario, joyas, entretenimiento, muebles, la medicina y en la educación.

2. Introducción a la red Blockchain.

Blockchain es una tecnología que utiliza un conjunto de registros compartidos por un grupo de computadoras; un bloque es un registro individual y la cadena son todos los registros que componen el libro mayor completo. La red está compuesta por computadoras que validan esos bloques y la cadena correspondiente a medida que ocurren las transacciones.

La base de datos creada se comparte entre los participantes de la red de manera transparente, por lo que todos pueden acceder a su contenido. La gestión de la base de datos se realiza de forma autónoma mediante redes *peer-to-peer* y un servidor de marcas de tiempo.

Estas transacciones podrían rastrear el "dinero" en forma de criptomoneda, o incluso podrían rastrear un vegetal desde la granja hasta su mesa (lo que puede, entre otras cosas, identificar la fuente de un brote de alguna enfermedad en nuestro suministro de alimentos más fácil que nunca).

Los objetivos de una cadena de bloques son proteger la integridad de los datos, pero hacerlo de manera descentralizada, a diferencia de una base de datos masiva controlada por una autoridad central.

Para implementar un modelo de desarrollo de *Blockchain* en entidades, es necesario comprender lo siguiente:

- Conceptos básicos
- Historia del *Blockchain*
- Características
- Ventajas y desventajas

2.1. Conceptos básicos

Se debe estar familiarizado con los siguientes términos:

Blockchain

El *Blockchain* es una cadena de bloques incorruptible donde cada bloque contiene datos de valor que son validados por todos los nodos de la red.

Cada bloque de la cadena incluye su valor *hash* y el del bloque anterior, que actúa como una huella digital única para que nadie pueda alterar los datos almacenados en él. La información almacenada en la cadena de bloques nunca se puede eliminar ni modificar. En cambio, es necesario agregar un nuevo bloque a la cadena para actualizar la información.

Descentralizado

Se dice que una cadena de bloques está descentralizada, ya que no se almacena en un solo lugar y no tiene un centro. En cambio, los datos guardados en *Blockchain* se distribuyen en muchas computadoras diferentes, llamadas nodos.

Dado que ninguna entidad tiene control sobre los datos, los usuarios interactúan entre sí directamente sin la participación de un tercero.

Consenso descentralizado

Una cadena de bloques es un sistema de igual a igual descentralizado que no tiene una autoridad central para controlar el intercambio de información. Aunque la participación de un administrador central no mantiene el sistema libre de corrupción, plantea las siguientes preguntas: ¿Cómo se toma una decisión en *Blockchain*? ¿Cómo se agrega una transacción a la cadena de bloques?

En un modelo centralizado normal, una autoridad central o una junta de tomadores de decisiones toman todas las decisiones necesarias. Pero no es posible en el caso de *Blockchain* ya que no tiene líder.

Los miembros de una red *Blockchain* deben llegar a un consenso a través de "mecanismos de consenso" para tomar decisiones. Los mecanismos de conceso es un problema de los sistemas distribuidos y consiste en poner de acuerdo múltiples procesos en un fin determinado.

Contratos inteligentes o *smart contracts.*

Los contratos inteligentes son los componentes básicos de las aplicaciones basadas en *Blockchain*. El concepto detrás de los contratos inteligentes es la gobernanza contractual de las transacciones entre dos o más participantes.

Se puede verificar mediante programación con la cadena de bloques, en lugar de una autoridad central. Además, los contratos inteligentes permiten a los usuarios controlar la propiedad al ofrecer una divulgación de datos controlada.

Minería

La minería se define como el proceso de agregar o validar transacciones al libro mayor distribuido. Se trata principalmente de crear un *hash* de un bloque que no se puede falsificar. Como resultado, protege la integridad de todo el sistema sin necesidad de un sistema central. Los mineros son los usuarios que utilizan el poder computacional para extraer bloques.

2.2. Historia de Blockchain

Podemos decir que *Blockchain* se inventó en 1991, la tecnología estaba bien equipada para mejorar la confianza digital dado el aspecto de descentralización que significaba que nadie jamás tendría el control de nada.

- 1991-2008: primeros años de la tecnología *Blockchain* [24]

Stuart Haber y W. Scott Stornetta eran jóvenes criptográficos que trabajaban en Bellcore (por sus siglas en inglés, *Bell Communications Research*). En el año de 1991 inventaron la técnica *Blockchain* para garantizar la integridad de los registros digitales por medio de una cadena

de bloques criptográficamente asegurada por la cual nadie podría alterar las marcas de tiempo de los documentos.

En el año 1992, actualizaron su sistema para incorporar árboles Merkle que mejoran la eficiencia y permitían así la recopilación de más documentos en un solo bloque. Pero fue hasta el año 2008 que *Blockchain* comenzó a ganar relevancia, gracias a Satoshi Nakamoto quien adoptó la técnica para implementar *Bitcoins*.

- Fase 1- Transacciones: 2008-2013: *Blockchain* 1.0: Aparición de *Bitcoins* [24]

Bitcoins nació en 2008 como la primera aplicación de la tecnología *Blockchain*. Satoshi Nakamoto, en su documento técnico publicado en el año 2009 lo detalló como un sistema electrónico *peer-to-peer.*

Nakamoto formó el bloque génesis, del cual se extrajeron otros bloques, interconectados dando como resultado una de las cadenas más grandes de bloques que transportan diferentes piezas de información y transacciones.

- Fase 2- Contratos: 2013-2015: *Blockchain* 2.0: Desarrollo de *Ethereum* [24]

Ethereum nació como una nueva cadena de bloques pública en 2013 con funcionalidades adicionales en comparación con *Bitcoins*, un desarrollo que ha resultado ser un momento crucial en la historia de *Blockchain*.

Vitalik Buterin programador y escritor ruso, diferenció *Ethereum* de *Bitcoins Blockchain* al habilitar una función que permite a las personas registrar otros activos, como contratos.
Lanzado oficialmente en 2015, *Ethereum Blockchain* ha evolucionado para convertirse en una de las mayores aplicaciones de la tecnología *Blockchain* dada su capacidad para admitir contratos inteligentes.

- Fase 3 Aplicaciones: 2015: *Hyperledger* [24]

En 2015, la fundación Linux dio a conocer el proyecto *Hyperledger Blockchain* de código abierto, que se centra en fomentar el uso de la tecnología *Blockchain* para mejorar el rendimiento y la confiabilidad de los sistemas actuales para respaldar las transacciones comerciales globales, actúa como desarrollo colaborativo de libros de contabilidad distribuidos.

- 2017: EOS.IO [24]

EOS, una creación de la empresa privada Block.one, detalla un nuevo protocolo de *Blockchain* impulsado por un EOS como Criptomoneda nativa. Por esa razón, EOS.IO se duplica como una plataforma de contrato inteligente, así como un sistema operativo descentralizado. Su objetivo principal es fomentar el despliegue de aplicaciones descentralizadas a través de una corporación autónoma descentralizada.

- 2018: *Blockchain* 3.0 [24]

En los últimos años, han surgido varios proyectos que aprovechan las capacidades de la tecnología *Blockchain* pero que también buscan solventar algunas de sus deficiencias y las de *Ethereum*, además de presentar nuevas características. Algunos de estos se detallan a continuación:

- NEO, lanzada en el 2014 como Antshares por Da Hongfei y Erik Zhang, considerada la primera plataforma de código abierto, descentralizada y *Blockchain* lanzada en China. NEO se presenta a sí mismo como el *Ethereum* chino.
- IOTA es una plataforma de Criptomoneda que está optimizada para el ecosistema de Internet de las cosas, ya que se esfuerza por proporcionar tarifas de transacción cero, 10 así como procesos de verificación únicos. También aborda algunos de los problemas de escalabilidad asociados con *Blockchain* 1.0 *Bitcoins*.
- Las cadenas de bloques Monero Zcash y Dash surgieron como una forma de abordar algunos de los problemas de seguridad y escalabilidad asociados con las primeras aplicaciones de la cadena de bloques. La plataforma de tres cadenas de bloques busca proporcionar altos niveles de privacidad y seguridad cuando se trata de transacciones.

- 2020: *Blockchain* y el futuro [24]

La tecnología *Blockchain* originalmente fue orientada a redes públicas, pero las grandes empresas están interesadas en hacer uso de esta tecnología para automatizar y mejorar sus procesos internos, lo que ha dado como resultado lo que se conoce como *Blockchain* privadas, híbridas y federadas.

2.3. Características

Las posibilidades y oportunidades que otorga *Blockchain* son diversas con una variedad de aplicaciones en distintas industrias, que otorgan mejoras de procesos y transformaciones en la forma en que se relacionan empresas y clientes.

Las transacciones entre dos partes interesadas se inician cuando uno de los participantes envía un mensaje sobre los términos y condiciones, a la red que gobierna la transacción *Blockchain*. Se pone en marcha un sistema de validación que evita el doble gasto de los miembros de la red y ellos juegan automáticamente el carácter de autenticadores para proteger y validar la transacción.

El estado de la transacción recientemente agregada, también se actualizará para los usuarios de la red, como un libro mayor público (registro de *Blockchain*) cuando se valida la transacción. Este mecanismo garantiza que las transacciones autorizadas no pueden ser modificadas posterior a ser validadas a través de algoritmos criptográficos, esto es lo que genera confianza entre los accionistas interesados en la tecnología *Blockchain*.

Como vemos *Blockchain* tiene mucho que aportar, de este proceso podemos extraer sus principales características:

- Descentralización
- Transparencia
- Inmutabilidad

Descentralización

Sin la ayuda de mediadores externos, *Blockchain* de manera transparente, puede transmitir la posesión de valiosos activos a todos los participantes y realizar transacciones donde los participantes están conectados en un mercado o plataforma de compra - venta.

Transparencia

Una trazabilidad completa de cada operación sin la necesidad de compartir datos privados. Facilita la rápida detección de cambios en la base de datos.

Inmutabilidad

Verificar la realidad de los datos de cada transacción en un tiempo específico en el bloque, es el mecanismo indiscutible proporcionado por la tecnología *Blockchain*. La posición, la historia y la propiedad de cada bloque no se puede modificar y se autentica automáticamente porque la información del bloque anterior está contenida en cada bloque de la cadena por medio de funciones criptográficas. Se puede añadir transacciones, pero no borrarlas.

Para tener una idea visual de una red básica de *Blockchain* y sus características nos apoyaremos de la imagen 2, extraída del libro *Blockchain: Aplicaciones y Entendimiento En El Mundo Real, Como el Blockchain Puede Ser Aplicado a Tu Mundo – Wayne Walker* [25]

Imagen 2: *Blockchain* y un vistazo a sus principales características

Fuente: Blockchain: Aplicaciones y Entendimiento En El Mundo Real. [online] Google Books Página 9

2.4. Ventajas y desventajas

En la siguiente tabla, se muestran las ventajas y desventajas relacionadas al *Blockchain* clasificadas por las propiedades principales observadas en esta tecnología.[26],[27]

Característica	Ventaja	Desventaja
Tecnología disruptiva	• Nunca ha existido una tecnología parecida con el potencial tan elevado de brindar seguridad y eficacia a los procesos. • Los aplicativos que se logren desarrollar tienen un alto valor económico que brinda competitividad e inversión de diferentes sectores.	• El desconocimiento generalizado sobre la aplicación de la tecnología. • La falta de suficiente personal idóneo que esté debidamente capacitado para la implementación.

Transacciones	• Las transacciones son posibles sin la confianza de un tercero como proveedor de servicio	• Cuando ocurre un problema no se sabe quién es el responsable de ello
Escalabilidad	• Es fácilmente establecido, conectado y expandido por fuente distribuida. • El costo de desarrollo del sistema es reducido.	• El número posible de transacciones que se pueden manejar posiblemente sea menor en comparación con la escalada de transacciones dentro de la economía real.
Transparencia	• Es posible acceder públicamente a todos los registros de transacciones y reducción de los costos de regulación.	• Dado que los detalles de las transacciones se revelan, todas las transacciones se pueden rastrear.
Seguridad	• El libro mayor es de propiedad conjunta (se mantiene íntegro), por lo que el costo relacionado con la seguridad se reduce	• Cuando se pierde la clave privada o es hackeada, no existe una solución general. • No proporciona confidencialidad.
Estabilidad del Sistema	• No existe un único punto de falla. • Si se producen errores o disminución de la función en ciertos sistemas participantes, el efecto en toda la red es muy leve	• Focalizada en grandes grupos mineros. • Es difícil de ejecutar en tiempo real el manejo de grandes volúmenes de transacción.

Tabla 2: Ventajas y desventajas del *Blockchain*

3. Elementos de una red *Blockchain*

Una vez se comprende que *Blockchain* no es una criptomoneda sino un registro descentralizado de información que se almacena en forma de transacciones que se agrupan en bloques, surgen preguntas como ¿Qué transacciones? o ¿Quiénes participan en dichas operaciones y ¿Cómo interactúan en la red? Dichas interrogantes se desarrollarán en este capítulo a continuación.

3.1. Consideraciones de una arquitectura y forma de operación de la red

De forma general se puede clasificar los diferentes tipos de *Blockchain* en cuatro grupos: públicos, privados, federados y "*Blockchain* como un servicio" (*Blockchain As a Service* por sus siglas en inglés). La distinción entre los tipos de *Blockchain* es el esquema de distribución y quién puede participar en el sistema. [28]

Las principales diferencias es el modelo de administración, nivel de descentralización o el grado de transparencia, entre otras características. Sin embargo, hay ciertas características que, en su mayoría, todos los *Blockchain*s tienen en común. Los tres elementos clave en la función del *Blockchain* son: sus participantes, activos, y transacciones [29]

3.1.1. Participantes

Son todos aquellos que tendrán un papel en la red *Blockchain*. Estos pueden ser desde las compañías que administran la red, usuarios, entidades auditoras, instituciones financieras, por ejemplo.

Para conocer el papel de cada participante, es necesario formular las siguientes interrogantes: ¿Cuáles son los permisos que tendrá sobre la red? ¿Cómo va a interactuar con el sistema? ¿Poseerá acceso a una copia de toda la cadena? ¿Logrará ver solo las transacciones en las que participe o poseerá acceso a más información? ¿Cuáles son las transacciones que podrá realizar? [30]

Dependiendo de las respuestas a estas preguntas, los participantes recibirán o no una copia de toda la cadena y tendrán o no permisos para ver y/o validar transacciones. Solo aquellos participantes que tienen una copia de la cadena son considerados nodos. El resto, que

accederán de forma general a través de un servicio web o una aplicación móvil, son simplemente usuarios.

3.1.2. Roles dentro de la red

La definición de un nodo puede variar significativamente. Con base al contexto en el que se utiliza, cuando se trata de redes informáticas o de telecomunicaciones, los nodos pueden ofrecer fines distintos, ya sea como un punto de redistribución o un punto final de comunicación. [31]

En términos simples, un nodo de red es un punto en el que se puede crear, recibir o transmitir un mensaje. Hay diferentes tipos, pero cada uno de ellos comparte una característica específica: necesitará hardware específico para alojar o simplemente conectarse a uno.

La tecnología *Blockchain* está descentralizada por naturaleza, una de las propiedades clave que la hizo tan atractiva para el público en general. Se basa en los principios de una red P2P (*Peer to Peer*). En la mayoría de las redes, no hay servidores dedicados, ni una sola autoridad, sino un consenso entre los usuarios. [32]

Como todos son cruciales para la seguridad y la integridad de la red, convertirse en miembro de una determinada comunidad de criptomonedas no solo es interesante, sino también una responsabilidad.

Por ejemplo, Bitcoin, tiene dos tipos de nodos. ***Full nodes,*** que almacenan una copia de la cadena de bloques y, por lo tanto, garantizan la seguridad y corrección de los datos en la cadena de bloques mediante la validación de los datos. El segundo tipo es un ***Lightweight nodes,*** cada usuario participante necesita conectarse a un nodo completo para sincronizarse con el estado actual de la red y poder participar. [31]

El objetivo de los nodos es conservar la confiabilidad de los datos almacenados en la cadena de bloques. La realidad es que se puede almacenar un historial completo de *Blockchain* con un solo nodo completo ejecutándolo. Cuantos más nodos tiene una cadena de bloques, más descentralizada se vuelve y, por lo tanto, se vuelve resistente a amenazas como fallas del sistema o cortes de energía. [33]

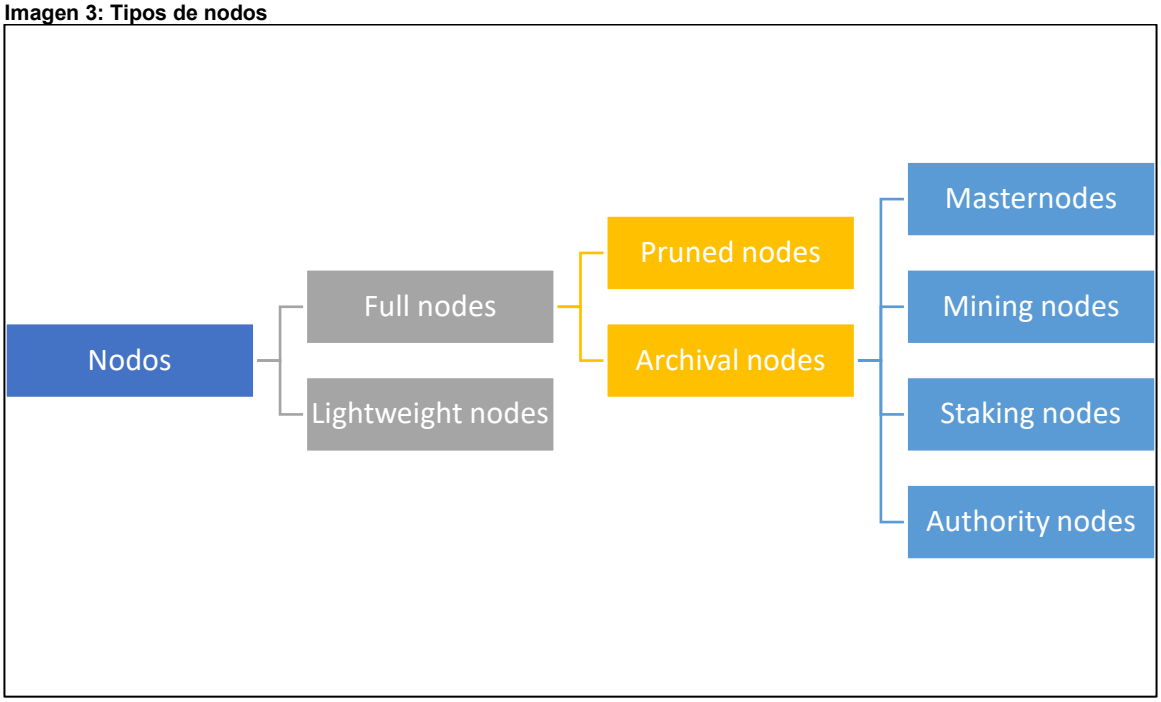
Cuando se agrega un nuevo dato (bloque) a una cadena de bloques, un nodo comunicará el bloque a otros nodos de la red. Según la validez del nuevo bloque y el tipo de nodo, los *full nodes* pueden rechazar o aceptar el bloque. Una vez que el nodo acepta un nuevo bloque, la información es almacenada y se guarda encima de los bloques preexistentes.

3.1.3. Tipos de nodos en Blockchain

En pocas palabras, hay dos tipos principales de nodos: *Full nodes* y *light nodes*. Otro término para describir los nodos es que existen clientes que proporcionan funciones de billetera. Los *full nodes* contienen una copia del historial de la cadena de bloques, incluidos todos los bloques creados. [34]

Los *light nodes* o nodos SPV (Por sus siglas en inglés, *Simple Payment Verification*) son billeteras que descargan solo los encabezados de los bloques y ahorran espacio en el disco duro para los usuarios. Exploremos los diferentes subtipos en detalle.

Imagen 3: Tipos de nodos



***Fuente*: Autoría propia**

Pruned nodes

La característica específica es que comienza a descargar bloques desde el principio y una vez que alcanza el límite establecido, elimina los más antiguos, conservando solo sus encabezados y la ubicación de la cadena.

Por ejemplo, si se establece un límite de tamaño de 500 MB, almacenará todos los bloques más recientes que puedan caber en ese espacio del disco duro, pero para llegar a ese estado, primero tendría que pasar por toda la cadena de bloques para validar todos. esos bloques anteriores. [35]

Los *pruned nodes* se consideran nodos completos y, por lo tanto, también pueden verificar transacciones y participar en el consenso[1].

Archival nodes

Son a lo que la mayoría de la gente se refiere cuando habla de nodos completos. Visualizan un servidor que aloja la cadena de bloques completa en su base de datos. La tarea principal es mantener el consenso y validar los bloques. La diferencia entre *pruned node* y *archival node* es, la cantidad de espacio en el disco duro que ocupan en su servidor o *PC.* [36]

Los *archival nodes* se pueden dividir en un par de subtipos: los que pueden agregar bloques a la cadena de bloques y los que tampoco pueden.

Nodos que pueden agregar bloques

Dependen de la aplicación de las reglas de consenso y requieren al menos un *archival node* completo para funcionar.

[1] El **consenso** o '*consensus*' no es más que la aceptación por todos los miembros de la red ***Blockchain*** de que la información que hay en la misma. La respuesta es: mediante un acuerdo entre los nodos de la red.

Mining nodes

Los mineros son en realidad nodos (ya sean *full node* o *light node*) que tienen como objetivo demostrar que han completado el trabajo requerido para crear un bloque. De ahí el nombre de consenso o prueba de trabajo (por sus siglas en inglés, *Proof of work)*.

Para completar la tarea, los mineros deben ser un nodo completo de archivo (*archival full node)* o recibir datos de otros nodos completos en la red para conocer el estado actual de la cadena de bloques y los parámetros requeridos para el siguiente bloque en línea.

Los participantes en el proceso emplean componentes de *hardware* (ya sean *CPU, GPU o ASIC*)[2] para resolver un problema criptográfico. La primera persona en completar la tarea transmite sus resultados a la red para que los nodos completos puedan verificarlos y, una vez que se logre el consenso, se le otorga el derecho de agregar un bloque a la cadena de bloques existente.

Por su trabajo, los mineros reciben una cantidad predefinida de monedas además de cualquier tarifa de transacción por el bloque. Esta cantidad de recompensa establecida se llama *coinbase* o una transacción de *coinbase*. Teniendo en cuenta que es la primera transacción del bloque, es gratuita, ya que el propio minero creó el bloque y lo incluyó.

Stakers nodes

Los *Stakers Nodes* se puede comparar a tener un depósito de dinero fiduciario tradicional. Usted compra monedas y las guarda, mientras que a cambio recibe un interés como recompensa.

Si bien existen diferentes opiniones sobre el mecanismo de consenso de la prueba de participación (por sus siglas en inglés, *Proof of work)*, la característica principal es que ganar dinero se puede comparar con participar en una lotería.

---

[2] **CPU:** Sigla de la expresión inglesa *central processing unit*, 'unidad central de proceso', que es la parte de una computadora en la que se encuentran los elementos que sirven para procesar datos.

GPU: Acrónimo de *Graphics Processing Unit* es un coprocesador dedicado al procesamiento de gráficos u operaciones de coma flotante, para aligerar la carga de trabajo del procesador central.

ASIC: *Application-specific integrated circuit* por sus siglas en inglés Un circuito Integrado para aplicaciones específicas es un circuito integrado hecho a la medida para un uso en particular

El objetivo final es determinar, en función de un conjunto predefinido de reglas y la posibilidad de suerte, quién será el próximo en crear un bloque y ser recompensado. Los factores incluyen la edad de las monedas (cuánto tiempo ha tenido sus monedas), cuántas tiene y su relación con las disponibles en la red.

Al participar, no necesita *hardware* costoso, solo mantiene su billetera criptográfica en línea las 24 horas del día, los 7 días de la semana, lo que se puede hacer con un dispositivo como la *Raspberry Pi.*[3]

Authority Nodes

Todos los nodos de *Blockchain* que se han revisado hasta este punto pueden unirse a una red y realizar sus tareas sin que nadie les dé permiso. Esa es la esencia de una cadena de bloques, su naturaleza descentralizada. Desafortunadamente, hay algunos inconvenientes en este enfoque y la solución implica emplear algún nivel de centralización para obtener beneficios como una mayor velocidad.

Los algoritmos de consenso incluyen Prueba de participación (por sus siglas en inglés, *Proof of work)*, delegada, tolerancia a fallas bizantinas[4], prueba de autoridad y otros. [32]

Las redes que utilizan dichos algoritmos deben definir un número fijo de nodos de autoridad. La comunidad vota sobre cuántos y quiénes serán, o el equipo de desarrollo lo define.

La tarea de estos nodos es, al igual que con los nodos completos, crear y validar bloques y, al mismo tiempo, distribuir información a los usuarios de la red. Todos los participantes, no elegidos para ser un nodo de autoridad, ejecutarán nodos ligeros (*light nodes,* por traducción en inglés) que dependen de los datos transmitidos para poder operar en la cadena de bloques.

[3] La Raspberry Pi es una serie de ordenadores de placa reducida, ordenadores de placa única u ordenadores de placa simple de bajo coste desarrollado en el Reino Unido por la *Raspberry Pi Foundation*, con el objetivo de poner en manos de las personas de todo el mundo el poder de la informática y la creación digital

[4] La tolerancia a faltas bizantinas es la resistencia de un sistema informático tolerante a faltas, en particular los sistemas informáticos distribuidos, a fallas de componentes electrónicos donde hay información imperfecta sobre si un componente falla

Masternodes

En comparación con los nodos completos, los *masternodes* por sí mismos no pueden agregar bloques a la cadena de bloques. Su único propósito es mantener un registro de transacciones y validarlas.

Ya sean *mining o stakers nodes*, son ellos los que escriben bloques en la cadena de bloques. Sin embargo, un beneficio adicional es que al ejecutar un *masternode,* no solo protege la red, sino que también puede ganar una parte de las recompensas por sus servicios. [31]

Para establecer un *masternode,* deberá bloquear una cierta suma de fondos como garantía. Se espera que esté en línea las 24 horas del día, los 7 días de la semana, y el alojamiento en un servidor privado virtual se considera una buena práctica.

Lightweight (SPV) Nodes

Otro tipo de nodos de *Blockchain*, que se utilizan en las operaciones de cifrado del día a día, es el nodo ligero (*light node)* o el nodo de verificación de pago simple (SPV, por sus siglas en inglés, *Simple Payment Verification*).

Estos tipos de nodos se comunican con la cadena de bloques mientras dependen de nodos completos para proporcionarles la información necesaria. Como no almacenan una copia de la cadena, solo consultan el estado actual para qué bloque es el último y transmiten las transacciones para su procesamiento.

Teniendo en cuenta las características anteriores, es evidente que la ejecución del nodo SPV (por sus siglas en inglés, *Simple Payment Verification*) no requiere muchos recursos, pero sacrifica la seguridad en aras de la conveniencia. [31]

Lightning nodes

Los *Lightning nodes* no se ajustan a las limitaciones de los nodos completos (*full nodes)* ni de los nodos ligeros (*light nodes)*.

La idea detrás de ellos es establecer una conexión entre usuarios fuera de la cadena de bloques. De esta manera, la carga en la red se reduce, los tiempos de transferencia se acortan significativamente y hay una mayor usabilidad de las criptomonedas. Las tarifas de transacción son realmente bajas en la red *lightning*, el equivalente de aproximadamente 10 a 20 satoshi[5] en general. [37]

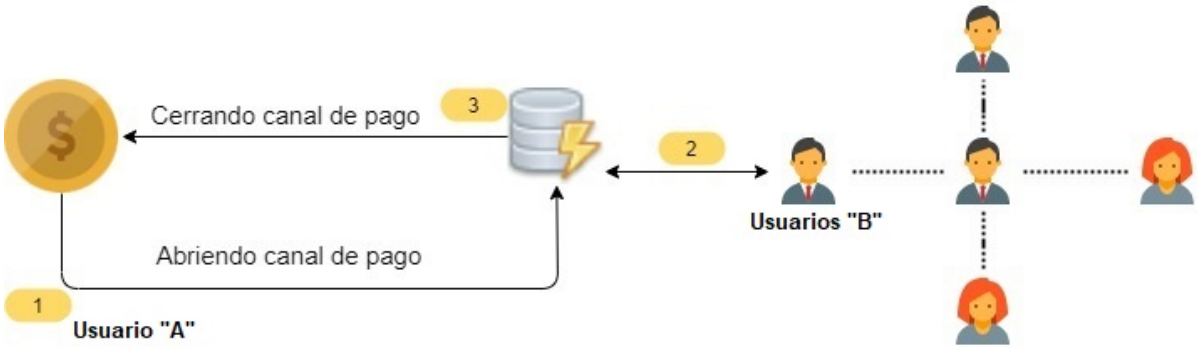
La forma en que funciona es abriendo un canal de pago separado entre entidades, por ejemplo, una tienda, usuario A y usuario B, comprador. El usuario A y B crean algo así como una caja de seguridad (una dirección con varias firmas) a la que ambos tienen llaves separadas. Usuario B, deposita sus fondos y los usa para pagar al usuario A.

Cada transacción es acordada por ambas partes y ocurre casi al instante. Una vez que haya realizado toda la transacción o simplemente se haya quedado sin dinero, él o la tienda pueden cerrar la conexión, tomar el último balance general y transmitirlo a la red.

De esta manera, en lugar de esperar a que se confirme cada transacción y llenar la red con datos que desperdician espacio, las partes pueden interactuar entre sí y reducir la carga en la cadena de bloques.

Además, si un tercero interactúa con la misma parte, la red *lightning* buscará un camino con la menor cantidad de intermediarios y las tarifas de transferencia más bajas, reduciendo así los tiempos de espera.

Imagen 4: ¿Cómo funciona un *Light node*?



***Fuente*: Autoría propia**

[5] Un satoshi es la unidad mínima de medida que se puede utilizar en el sistema Bitcoin. Es la fracción más pequeña en la que puedes dividir un bitcoin.

3.1.4. Activos e información

De acuerdo al Banco Interamericano de Desarrollo (por sus siglas en ingles *Inter-American Development Bank)* define activos de información de la siguiente manera:

"…La forma de entender este grupo es pensar que cuando los participantes hacen una transacción, en muchas ocasiones están transfiriendo algo. Ese "algo" es el activo, y puede ser un documento, un certificado, un informe, un token, una moneda digital, etc."

Se puede encontrar una definición de activos o activos virtuales en la "*Ley para Regular a las Instituciones de Tecnología Financiera"* ley aprobada en la 81 Convención Bancaria en Acapulco, México, denominada, la "Ley Fintech", establece que un activo virtual es:

"...*la representación de valor registrada electrónicamente y utilizada entre el público como medio de pago para todo tipo de actos jurídicos, y cuya transferencia únicamente puede llevarse a cabo a través de medios electrónicos."* [38]

También, el Banco de México lo define como *"(i) una unidad de información que no representa la tenencia de algún activo subyacente a la par, y que es unívocamente identificable, incluso de manera fraccional, almacenada electrónicamente"* [39]

De acuerdo a la definición, se considera que el valor está definido por la oferta y la demanda, que depende únicamente de los valores agregados en materia de ciberseguridad y operatividad del activo virtual, que pueda dar el medio (activo) electrónico a los usuarios.

Dentro del concepto, no se tomará en cuenta a aquellos activos que utilizan la misma tecnología, por ejemplo, si se habla de un activo virtual podemos mencionar la tecnología *Blockchain* como activo virtual, pero las criptomonedas utilizan la misma tecnología para funcionar, la tenencia a la par de algún activo subyacente como por ejemplo una moneda de curso legal no se tomara en cuenta dentro del significado, porque no existe un bien que respalde el valor de un activo virtual uno a uno.

El funcionamiento de los activos virtuales se maneja por medio de un dominio el cual permite la emisión de una cantidad aproximada de monedas a través de diferentes transacciones electrónicas donde su almacenamiento se identifica por medio de llaves o códigos los cuales identifican la aparición de lo virtual o la utilización de cada uno de ellos.

3.1.4.1. Características de los activos virtuales

Característica	Activo virtual
Posesión	Por medio de una llave privada que permite iniciar transferencias dentro del registro distribuido de dicho activo.
Almacenamiento	Es digital, se almacena en los nodos de la red distribuida dentro del activo virtual.
Medio de intercambio	Mediante un mensaje que se envía a la red del activo virtual.
Ataques de ciberseguridad	Se pueden evitar mediante la validación de tenencia a partir de la revisión del registro que se ha distribuido y que contiene el histórico de las transacciones, el proceso de minado también es fundamental en estos casos.
Respaldo	Actualmente el respaldo está basado en la confianza de los usuarios en la red del activo virtual.

Fuente: Banco Central de México. [12] [7] 34 Y 39

Tabla 3: Característica de los activos virtuales

3.1.5. Transacciones

Las operaciones que se realizan para agregar información a una red *Blockchain* son denominadas transacciones, también conocidas como TX. [40] [41]

Las transacciones en una cadena de bloques representan transiciones de estado autorizadas. Su contenido puede ser muy variado y generalmente depende del tipo de red que se esté operando, puede registrar datos y transferir el control de los activos digitales entre los participantes de la misma red. [42]

Las Criptomonedas son un tipo de activo digital, pero otros tipos de *tokens* de activos digitales también se pueden implementar en *Blockchain*. Las transacciones de criptomonedas, toman forma de operaciones de compra-venta de activos o servicios.

¿Cómo funcionan las transacciones en una red *Blockchain*?

En este punto nos apoyaremos del Ciclo de Vida de las Transacciones, extraído del libro *"Architecture for Blockchain Applications".* Donde se observan las seis etapas del flujo de una transacción.

El primer paso es la creación, donde el usuario A, decide transferir un activo al usuario B. Para cada transacción es necesario que el usuario utilice su monedero o *wallet*, que no es más que una aplicación que le permite administrar sus fondos, realizar o recibir transacciones que se originen en una determinada *Blockchain*.[43]

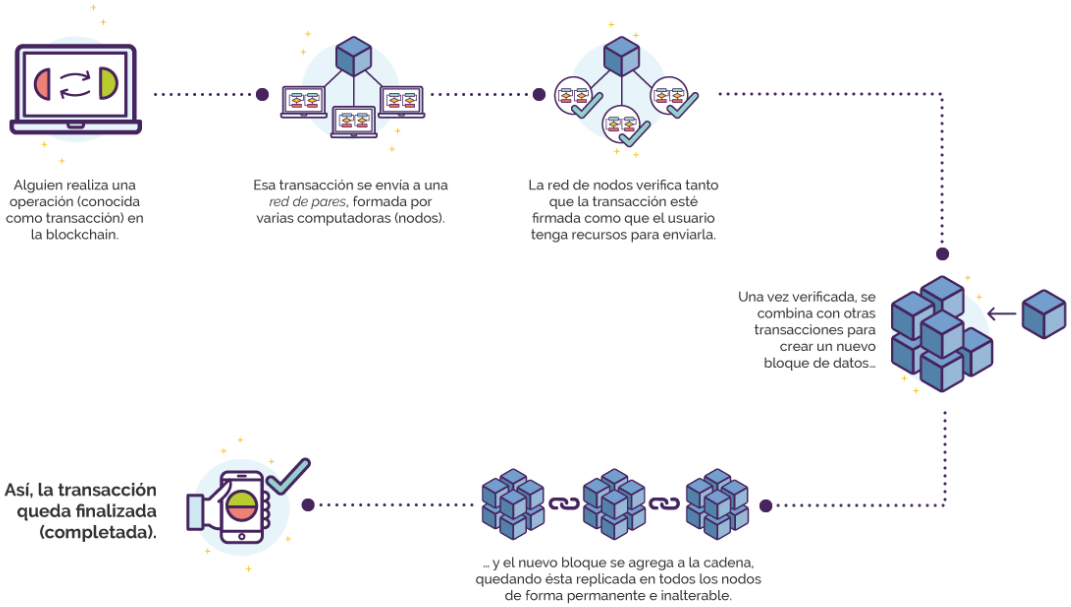
Una vez creada, la transacción se firma digitalmente con la firma del usuario A, esta debe contener el *hash* de la transacción anterior y la clave pública del usuario B. Con esto proporciona la autorización para gastar el dinero, crear un contrato o pasar los parámetros de datos asociados con las transacciones. [43]

La siguiente etapa es la validación, la red *Blockchain* se encarga de verificar la validez de la transacción. Las transacciones no válidas se descartan. Las transacciones válidas que el nodo desconocía anteriormente se propagan a otros nodos conectados.

Durante la etapa de propagación, los remitentes no necesitan confiar en ningún nodo individual al que envían la transacción, siempre que la envíen a suficientes otros nodos. Los destinatarios no necesitan confiar en los remitentes, porque todas las transacciones están firmadas y pueden ser validadas de forma independiente por cualquier nodo. [43]

Cuando la transacción se ha propagado lo suficiente, (esto requiere de solo pocos segundos) llega a un nodo de "minería", se verifica y puede incluirse en un bloque. Cuando ese bloque se agrega a la cadena, la transacción queda incorporada definitivamente y se considera como completada o confirmada y el usuario B recibe en su *wallet* el monto enviado por el usuario A.

Imagen 5: Ejemplo de una transacción en la red Blockchain [42]



***Fuente*: https://bfa.ar/blockchain/bloques-y-transacciones**

En todas estas etapas, se hacen uso de elementos que robustecen la seguridad de la red *Blockchain*. La criptografía de clave pública y las firmas digitales, se utilizan normalmente para identificar las cuentas y garantizar la integridad y autorización de las transacciones iniciadas, en una cadena de bloques.

Una transacción firmada, debe contener toda la información necesaria para su ejecución y para poder verificarla en la cadena de propiedad con apoyo de las marcas de tiempo.

La red tiene como objetivo llegar a un consenso sobre el último bloque que se incluirá en la cadena de bloques. Existen diferentes mecanismos de consenso, por ejemplo "*Proof of Work " o " proof of stake*", descritos con anterioridad.

Elementos de una Transacción [41]

Entradas *(inputs)*. Las entradas son las referencias a una salida de una transacción pasada, que no ha sido empleada en ninguna otra transacción. Estas permiten confirmar la procedencia de los activos que se utilizarán en una transacción. Y son las que contienen la dirección donde originalmente se recibieron las criptomonedas.

Salidas *(outputs)*. Estas contienen la dirección a la cual se realiza la transferencia y la cantidad enviada. Además, contienen las direcciones de cambio o de retorno donde son enviadas las vueltas de las transacciones. Por lo que en una transacción puede contener más de una salida.

Identificador (TXid). Cada transacción realizada tendrá su propio *hash*. Este *hash* se genera a partir de las entradas y las salidas. Este valor es el que permite identificar una transacción de forma única e irrepetible dentro de una *Blockchain*.

Tarifa de comisión *(fee)*. Las *fee* es el pequeño pago que reciben los mineros por procesar una transacción. Así, el minero que genere un nuevo bloque, recibirá una *fee* por cada transacción procesada dentro de dicho bloque. La comisión no viene de forma explícita en el contenido de una transacción, es decir no se asocia a ninguna salida.

Ventajas de las transacciones con *Blockchain* [41]

Rapidez, la transferencia de activos por medio de una red *Blockchain* se completa en segundos.

Irreversibilidad, una vez que se realiza una transacción y ésta se añada a la *Blockchain*, es prácticamente imposible de revertir o modificar.

Seguridad, las transacciones con *Blockchain* se realizan empleando direcciones públicas y claves privadas, lo que permiten enviar o recibir transacciones de criptomoneda sin riesgos de robo alguno.

Comisiones más económicas, las *fees* canceladas a los mineros para que procesen una transacción, son realmente bajas, en comparación con los porcentajes solicitados por los bancos u otros sistemas intermediarios tradicionales.

3.1.6. Aprobaciones y algoritmos de consenso

Como se comentó anteriormente, la red *Blockchain* tiene como objetivo llegar a un consenso sobre el último bloque que se incluirá en la cadena de bloques. Pero ¿cómo se puede lograr,

que los diferentes miembros de una red descentralizada, se pongan de acuerdo? Para solventar eso se hace uso de los Algoritmos de Consenso.

Como definición podemos decir que algoritmo de consenso, es el método previamente definido por un grupo para la toma de decisiones. Es una forma de resolución donde los miembros del grupo, deben apoyar la decisión mayoritaria, asegurando la igualdad, equidad y el beneficio de todos. [44]

Podemos decir que los algoritmos de consenso son el núcleo de la arquitectura *Blockchain*. Estos algoritmos son los que tomarán las decisiones y resolverán cualquier discrepancia asegurando la continuidad de la red.

3.1.6.1. El Problema de los Generales Bizantinos

Antes de presentar los algoritmos de consenso, es necesario conocer un poco de historia, que hará entender como la tecnología *Blockchain* resolvió un problema planteado en los años 70's, sobre la teoría computacional de los sistemas distribuidos.

El problema plantea un escenario de guerra, en el que un grupo de "n" generales bizantinos que están asediando una ciudad desde distintos lugares, tienen que ponerse de acuerdo para atacar o retirarse de forma coordinada.

Entre los generales hay solo uno que puede cursar la orden por ser el comandante, el resto se dice que son tenientes y es posible que algunos sean desleales. La importancia de este problema no es si deciden atacar o no, sino el lograr llegar a un consenso.

Fue hasta en el año 1982 cuando Leslie Lamport, Robert Shostak y Marshall Pease proporcionaron distintos algoritmos de solución en función de condiciones adicionales. Entre esas soluciones se encontraba el algoritmo SM (donde SM viene del inglés *Signed Messages*). [45]

Las condiciones son que todos se pueden comunicar con todos y que los mensajes son firmados. Según este algoritmo los generales no recibirán más mensajes cuando tengan todas las posibles combinaciones. Una vez recibidas, cada nodo toma la decisión basándose en la orden transmitida por la mayoría. En este escenario los comandantes traidores son descubiertos inmediatamente ya que han firmado órdenes contradictorias.

Satoshi Nakamoto, puso en práctica esta teoría y dio vida a la red *Blockchain* de Bitcon, agregó servidores de estampa del tiempo e implementó el método de consenso *Proof of Work,* logrando la transmisión de dinero sin un intermediario confiable. [46]

Prácticamente todo sistema distribuido necesita implementar algoritmos de consenso, para resolver resultados contradictorios y obtener la mejor salida.

3.1.6.2. Diferentes tipos de algoritmo de consenso

Los algoritmos o métodos de consenso han evolucionado con el paso del tiempo, adaptándose a las necesidades y características de cada red *Blockchain*, pero siempre con el mismo objetivo, proveer equidad e integridad en la red.

A continuación, se enlistarán los algoritmos de consenso existentes, y profundizaremos solo en aquellos que consideramos relevantes para el desarrollo de esta investigación. Al finalizar se hará una comparativa entre ellos.

- *Proof of Work* (*PoW*) - Prueba de Trabajo
- *Proof of Stake* (PoS) - Prueba de Participación
- *Delegated Proof of Stake* (DPoS) - Prueba de Participación Delegada
- *Leased Proof of Stake* (LPoS) - Prueba de participación arrendada
- *Proof of Elapsed Time* (PoET) - Prueba de tiempo transcurrido
- *Practical Byzantine Fault Tolerance* (PBFT) - Práctica de tolerancia a faltas bizantinas.
- Tolerancia a faltas bizantina simplificada
- Tolerancia a faltas bizantina delegada
- Grafo de acíclico dirigido
- *Proof-of-Activity* (PoA) - Prueba de Actividad
- *Proof-of-Importance* (PoI) - Prueba de Importancia
- *Proof-of-Capacity* (PoC) - Prueba de Capacidad
- *Proof-of-Burn* (PoB) - Prueba de Quemado
- *Proof-of-Weight* (*PoW*) - Prueba de Peso

Proof of Work (*PoW*) - Prueba de Trabajo [44]

El consenso de Prueba de Trabajo consiste en que cada nodo complete una ecuación extremadamente compleja para terminar cada bloque.

El propósito de la complejidad de la ecuación, es garantizar que cada nodo se vea obligado a ejercer una cantidad significativa de potencia de procesamiento y electricidad para resolverlo.

Mientras más poder de procesamiento tenga un minero, más probable será que pueda resolver correctamente y más rápido la compleja ecuación y gane la recompensa del bloque.

La arquitectura *Blockchain* es inteligente y al percibir que tiene un ataque *DDoS*, se prepara para encontrarlo y el sistema de consenso requiere de muchos más cálculos como manera de protección. Aquí es donde los mineros son útiles y el sistema se apoya con las funciones *hash*.

Los nuevos bloques vienen con la función *hash*, y cada uno de ellos contiene la función *hash* del bloque anterior. De esta manera, la red agrega una capa adicional de protección y previene cualquier tipo de violación.

La única forma de controlar una red *Blockchain* seria poseer más del 51% de los *ledgers*, que sería lo mismo que poseer el 51% de la potencia de procesamiento total, lo que hace extremadamente difícil que la red sea vulnerada.

Ventajas y Desventajas de *Proof of Work* (*PoW*)

Ventajas

- Algoritmo seguro, utiliza codificación *hash* de alta seguridad.
- Es fácil de implementar.
- Se adapta fácilmente a distintos hardware (*ASIC, GPU, CPU*)
- Capacidad de resistencia al ataque de Denegación de Servicio (*DDoS*)

Desventajas

- Necesita grandes recursos para ser mantenido enérgico y computacionalmente. A modo de comparación, una transacción que utiliza el consenso *PoW* consume la cantidad de energía que usa un hogar promedio en un periodo de 24 horas.
- El método de consenso *PoW* generalmente tiene tiempos de confirmación de transacción más lentos en comparación a otros métodos de consenso.
- Centralización de mineros. Es posible que un fabricante o un grupo de fabricantes de Bitcoin se unan y crezca en gran medida, por lo que demandaría mucha más energía y puede tratar de crear nuevas reglas en el sistema de minería. Lo que amenazaría la estructura descentralizada de la red.

***Proof of Stake* (*PoS*) - Prueba de Participación [44]**

El algoritmo de prueba de participación fue desarrollado en el año 2011 y utilizado por primera vez con la criptomoneda *Peercoin* en el año 2012. [47]

El método *PoS* trata de eliminar las desventajas del consenso *Proof of Work* (*PoW*), aquí no existen nodos mineros y tampoco la necesidad de calcular y resolver ecuaciones complejas, en cambio todos los participantes de la red tienen la oportunidad de validar nuevos bloques, pero tiene más oportunidad aquellos que posean más monedas.

Todos los mineros de la red son elegidos aleatoriamente. Si tienen una cantidad específica de monedas almacenadas previamente en su billetera, entonces estarán calificados para ser un nodo en la red.

Posteriormente para calificar para ser minero es necesario realizar un tipo de apuesta, se debe presentar una cantidad de monedas, las cuales quedan bloqueadas en una caja fuerte virtual, si la red detecta irregularidades, el participante pierde esas monedas como penalidad.

Cuantas más monedas apueste un nodo, mayor será la posibilidad de que se seleccione para crear el siguiente bloque.

Hay otras formas de participar en la apuesta, si la cantidad apostada es muy alta, el participante puede unirse a un grupo o pool y obtener ganancias a través de ello.

Esto también significa que los nodos que intentan vulnerar el sistema tienen más que perder. Prácticamente este es el mecanismo de protección que utiliza el método *Proof of Stake* (*PoS*), sin la necesidad de un enorme gasto energético.

El proceso es bastante simple, los nuevos bloques se crearán de forma proporcional al número de monedas en función de la cartera. Por ejemplo, si el participante seleccionado posee el 10% de todas las monedas, podrá minar un 10% de bloques nuevos.

Hay muchas tecnologías de *Blockchain* que usan una variante de este método. De igual forma, todos los algoritmos funcionan de la misma manera para minar nuevos bloques. Cada minero recibirá una recompensa por el bloque al igual que una parte de las tarifas de transacción.

Ventajas y Desventajas de *Proof of Stake* (*PoS*) [44]

Ventajas

- No requiere de un hardware especial ni se gastan recursos en resolver cálculos matemáticos, que añadan dificultad. Solo necesita un sistema informático funcional y una conexión a Internet estable.
- El *Blockchain* basado en el algoritmo de consenso prueba de participación es mucho más eficiente con respecto a la energía.
- La probabilidad de un ataque del 51% se reduce ya que se necesitaría previamente tener una gran cantidad de monedas.
- La seguridad es un compromiso demostrado por cada nodo de la red.
- Control del tiempo para la creación del bloque. Mientras con el método *PoW*, los nodos compiten para solucionar un problema, en *PoS* la asignación es aleatoria.

Desventajas

- Escalabilidad.
- Lentitud en las transacciones por la toma de decisiones
- El anonimato es más difícil de mantener, ya que se realiza un proceso de identificación para poder participar.

- Cuantas más participaciones se posea, mayor poder se tiene en la red, para la toma de decisión.

***Practical Byzantine Fault Tolerance* (PBFT) - Práctica de tolerancia a faltas bizantinas.**

La práctica de la tolerancia a faltas bizantinas en inglés *Practical Byzantine Fault Tolerance* (PBTC), se centra en máquina de estados. El algoritmo está diseñado para sistemas de consensos asíncronos y optimizados de una forma más eficiente. Se selecciona un nodo como el principal y otros funcionan como respaldo. [44]

El nivel de comunicación es bastante alto, esto se debe a que quieren verificar toda la información que sea encontrada en la red. Esto elimina el problema de la información poco confiable.

Para este fin, se deben ejecutar tres tipos de protocolos básicos: [48]

- **Acuerdo de coherencia:** cómo llegar a un consenso
- **Protocolo de punto de control:** punto de restauración similar al sistema operativo
- **Ver protocolo de reemplazo:** cada nodo de servidor del sistema funciona con la misma información de configuración, que se denomina "vista". La información de configuración está determinada por el nodo maestro. Cuando se reemplaza el nodo maestro, la vista cambia en consecuencia.

El sistema de consenso *PBFT*, generalmente supone que el número de nodos defectuosos es "m", y el número total de nodos de servicio es 3m + 1. La solicitud de cada cliente debe pasar por 5 etapas, utilizando dos pares de dos interacciones, la solicitud del cliente se ejecuta después de que el servidor está de acuerdo.

El proceso básico de todo el acuerdo es el siguiente: [48]

- El cliente envía una solicitud para activar la operación de servicio del nodo maestro.
- Después de que el nodo maestro recibe la solicitud, inicia un protocolo trifásico para transmitir la solicitud a cada nodo esclavo.
- En la etapa de asignación de número de secuencia, el nodo maestro asigna un número de secuencia "n" a la solicitud, difunde el mensaje de asignación de número de secuencia

y el mensaje de solicitud del cliente "m", y construye un mensaje PRE-PREPARE a cada nodo esclavo.

- En la fase de interacción, el mensaje PRE-PREPARE se recibe del nodo y el mensaje PREPARE se transmite a otros nodos de servicio.
- En la fase de confirmación del número de serie, después de que cada nodo verifica la solicitud y el pedido en la vista, emite un mensaje de COMPROMISO, ejecuta la solicitud del cliente recibido y responde al cliente.
- El cliente espera respuestas de diferentes nodos. Si las respuestas "m" + 1 son las mismas, la respuesta es el resultado de la operación.

Ventajas y Desventajas de *Practical Byzantine Fault Tolerance* (*PBTC*) [44]

Ventajas

- No se requiere confirmación. Si los nodos concuerdan con un bloque específico, este se finaliza. Esto se debe al hecho de que todos los nodos auténticos se comunican entre sí al mismo tiempo y llegan a un entendimiento del bloque especificado.
- Reducción de consumo de energía en comparación con el método *PoW*. En este modelo no todos los mineros están resolviendo el típico algoritmo de *hashing*, lo que también reduce la necesidad de un alto poder computacional.

Desventajas

- Escalabilidad. El metodo PBFT requiere de un alto nivel de comunicación entre todos sus nodos, si el grupo de nodos aumenta en gran medida, el sistema puede tener dificultades para realizar un seguimiento de todos los nodos y no podrá comunicarse con cada uno de ellos.
- PBFT es vulnerable a Ataque Sybil. En estos ataques, se pueden manipular un grupo de nodos, y al hacerlo, estos comprometen toda la red.

Comparación entre los Algoritmos de Consenso estudiados [44]

Algoritmos de consenso	Plataforma *Blockchain*	Lanzado desde	Lenguajes de programación	Contrato inteligente	Pros	Contra
PoW	Bitcoin	2009	C++	No	Menos oportunidades de un ataque de 51% Mejor seguridad	Gran consumo de energía Centralización de mineros
PoS	NXT	2013	Java	Si	Ahorro de energía Mayor descentralización	Problema de nada que perder
PBFT	Hyperledger Fabric	2015	JavaScript, Python, Java REST and Go	Si	Sin necesidad de confirmación Ahorro de energía	Brecha de comunicación Ataque Sybil

Tabla 4: Comparación entre los Algoritmos de Consenso

3.1.7. Contratos Inteligentes o *Smart Contracts*

3.1.7.1. Definición y usos de los contratos inteligentes

Los contratos inteligentes han surgido como un nuevo caso de uso prometedor de la *Blockchain* que amplía sus horizontes y la convierte en una plataforma informática distribuida.

En sentido general, los *smart contracts* son contratos autoejecutables donde los usuarios pueden codificar sus acuerdos y relaciones de confianza, que luego se almacena en una cadena de bloques de alojamiento. [49]

Los contratos inteligentes pueden facilitar actividades comerciales y de intercambio de forma segura y confiable, al proporcionar transacciones automatizadas sin la supervisión de un sistema financiero externo como bancos, tribunales o notarios. Estas transacciones son rastreables, transparentes e irreversibles. [49]

En un sentido formal, podemos definir a un *smart contracts* de la siguiente manera:

"Cualquier acuerdo en el que se formalicen todas o algunas de sus cláusulas mediante Scripts o pequeños programas, cuyo efecto sea que, una vez concluido el acuerdo y señalados uno o varios eventos desencadenantes, la producción de los eventos programados conlleva la ejecución automática del resto del contrato, sin que quepa modificación, bloqueo o inejecución de la prestación debida". [50]

En un sentido técnico, podemos definir un *smart contracts* de la siguiente forma:

"Los smart contracts son programas implementados como datos en el libro mayor de Blockchain y ejecutados en transacciones en Blockchain. Pueden contener y transferir activos digitales administrados por Blockchain y puede invocar otros contratos inteligentes almacenados en la cadena de bloques. En un smart contract, el código es determinista e inmutable una vez implementado" [43].

Las aplicaciones posibles a este modelo de contratación son muy variadas. Se puede usar *Smart contracts* en múltiples transacciones, como [50]:

- **Préstamos:** si el deudor no efectúa un pago, el contrato automáticamente podría revocar las claves digitales que le dan acceso a los fondos o activar las garantías.
- **Depósitos en garantía:** compras por internet, verificada la entrega (registro del código de barras en destino, seguimiento del documento electrónico de trazabilidad, huella digital del receptor entre otros) se libera el pago.
- **Controles de gasto:** liberación de subvenciones y/o pagos a proyectos, previa entrega de certificados.

- **Herencias y donaciones:** liberación de los fondos, legados etc. ante el registro del certificado de defunción.
- **Piscinas de voto multifirma *(multi-signature voting pools)*:** se efectúa un depósito en una parte de confianza para garantizar el cumplimiento de una transacción, sin que ninguna de las partes tenga acceso al mismo hasta que dos o más de las partes señaladas en el acuerdo aprueben la transacción, liberando con ello el depósito en favor de la persona indicada como cumplidora o beneficiaria de la prestación bloqueada.

  El servidor que ejecuta el software nunca recibe fondos de los usuarios, ni transmite los fondos de los usuarios, ni puede acceder a sus fondos, ni tiene la posibilidad de cambiar el saldo de un usuario, revertir una transacción, o confiscar el dinero. Sólo puede retenerlo o liberarlo válidamente.
- **Dobles depósitos:** De forma parecida a la anterior, sólo que se elimina a los terceros como fuente de verificación. Las partes, comprador y vendedor realizan una transacción de depósito vinculada a un contrato inteligente.

  El programa tiene un tiempo determinado. Si las partes no cumplen lo programado el dinero se transfiere a una tercera parte, se “quema” en alguna dirección de la que no tienen clave privada de acceso, por lo que hay un fuerte incentivo para cumplir en plazo y liberar el depósito.

3.1.7.2. Lenguajes de Programación para desarrollo de *Smart Contracts*

Los contratos inteligentes son generalmente escritos con lenguajes específicos de dominio (DSL por sus siglas en inglés *Domain Specific Languages*), como *Solidity* en Ethereum, Pact en Kadena, Liquidity en la plataforma Tezos.

En algunos casos, lenguajes de propósito general como Kotlin, Go y Java también se utilizan para escribir contratos inteligentes, principalmente debido a la familiaridad y razones de usabilidad para los desarrolladores. [49]

En este apartado se aborda el listado de los lenguajes de programación más utilizados para desarrollo de contratos inteligentes para redes *Blockchain* [51]

- **C ++**

C ++ es un lenguaje de programación de propósito general que comprende al menos más de 4.4 millones de desarrolladores. Su mayor fortaleza es la capacidad de escalar aplicaciones que consumen muchos recursos y hacer que funcionen sin problemas.

Como la cadena de bloques *EOS* admite contratos inteligentes a través de su máquina virtual *WebAssembly*, cualquier lenguaje que pueda compilarse en *Web Assembly (WASM)* podrá programar contratos inteligentes. Sin embargo, C ++ es el lenguaje recomendado para que los desarrolladores lo utilicen en EOS.

Imagen 6: Ejemplo C++

```C++
#include <eosio/eosio.hpp>
class [[eosio::contract]] hello : public eosio::contract {
  public:
      using eosio::contract::contract;
      [[eosio::action]] void hi( eosio::name user ) {
         print( "Hello, ", user);
      }
};
```

***Fuente*: https://developers.eos.io/welcome/latest/getting-started-guide/hello-world**

- **Solidity**

Solidity fue desarrollado por primera vez por Gavin Wood, Yoichi Hirai, Christian Reitweissner y muchos otros contribuyentes principales de Ethereum para ayudar a desarrollar contratos inteligentes.

Solidity es el principal lenguaje de programación de contratos inteligentes que se utiliza para crear contratos en la cadena de bloques Ethereum. Es un lenguaje de programación de alto nivel que se parece a Python, C ++ y JavaScript y es orientado a contratos, lo que significa que los contratos inteligentes tienen la responsabilidad de almacenar toda la lógica de programación que realiza transacciones con la cadena de bloques.

El lenguaje de programación Solidity se ejecuta en la máquina virtual Ethereum (EVM, por sus siglas en inglés Ethereum Virtual Machine) que está alojada en los nodos Ethereum

conectados al *Blockchain*. Admite herencia, bibliotecas y mucho más y está tipado estáticamente. Es capaz de construir aplicaciones *Blockchain* que aumentan la fuerza industrial.

Imagen 7: Ejemplo Solidity

Show less | Copy

```
// SPDX-License-Identifier: GPL-3.0
pragma solidity >= 0.7.0;

contract Coin {
    // The keyword "public" makes variables
    // accessible from other contracts
    address public minter;
    mapping (address => uint) public balances;

    // Events allow clients to react to specific
    // contract changes you declare
    event Sent(address from, address to, uint amount);

    // Constructor code is only run when the contract
    // is created
    constructor() {
        minter = msg.sender;
    }

    // Sends an amount of newly created coins to an address
    // Can only be called by the contract creator
    function mint(address receiver, uint amount) public {
        require(msg.sender == minter);
        require(amount < 1e60);
        balances[receiver] += amount;
    }

    // Sends an amount of existing coins
    // from any caller to an address
    function send(address receiver, uint amount) public {
        require(amount <= balances[msg.sender], "Insufficient
balance.");
        balances[msg.sender] -= amount;
        balances[receiver] += amount;
        emit Sent(msg.sender, receiver, amount);
    }
}

```

***Fuente*: https://ethereum.org/nl/developers/docs/smart-contracts/languages/**

- **JavaScript**

  JavaScript es un lenguaje de programación orientado a objetos que es dinámico y ligero. JavaScript fue creado por Brendan Eich. Junto con HTML y CSS, JavaScript forma los tres pilares del diseño web, no obstante JavaScript se utiliza para crear contratos inteligentes en la cadena de bloques NEO.

  NEO es una plataforma *Blockchain* que facilita el desarrollo de contratos inteligentes y activos digitales. La palabra NEO se origina en el idioma griego antiguo y se traduce como 'nuevo', 'moderno' y 'joven'. NEO tiene como objetivo utilizar contratos inteligentes para convertirse en una plataforma descentralizada, digital y distribuida para activos no digitales. Su objetivo específico es convertirse en una alternativa digital para las transferencias de activos que actualmente no son digitales.

  NEO tiene como objetivo brindar opciones y libertad a los desarrolladores. Como admite una variedad de lenguajes de programación convencionales, muchos desarrolladores pueden escribir contratos inteligentes en NEO y desarrollar y realizar sus propias ideas.

Imagen 8: Ejemplo JavaScript

```
const path = require('path');
const fs = require('fs-extra');
const solc = require('solc');

const sourceFolderPath = path.resolve(__dirname, 'contracts');
const buildFolderPath = path.resolve(__dirname, 'build');


const getContractSource = contractFileName => {
    const contractPath = path.resolve(__dirname, 'contracts', contractFileName);
    return fs.readFileSync(contractPath, 'utf8');
};

let sources = {};

var walk = function (dir) {
    var results = [];
    var list = fs.readdirSync(dir);
    list.forEach(function (file) {
        file = dir + '/' + file;
        var stat = fs.statSync(file);
        if (stat && stat.isDirectory()) {
            results = results.concat(walk(file));
        } else {
            if (file.substr(file.length - 4, file.length) === ".sol") {
                sources = {
                    ...sources,
                    [file]: {
                        content: getContractSource(file)
                    }
                };
            }
            results.push(file);
        }
    });
    return results;
};
walk(sourceFolderPath);

const input = {
    language: 'Solidity',
    sources,
    settings: {
        outputSelection: {
            '*': {
                '*': ['*']
            }
        }
    }
}

console.log('\nCompiling contracts...');
const output = JSON.parse(solc.compile(JSON.stringify(input)));
console.log('Done');

let shouldBuild = true;

if (output.errors) {
    console.error(output.errors);
    // throw '\nError in compilation please check the contract\n';
    for (error of output.errors) {
        if (error.severity === 'error') {
            shouldBuild = false;
            throw 'Error found';
            break;
        }
    }
}

if (shouldBuild) {
    console.log('\nBuilding please wait...');

    fs.removeSync(buildFolderPath);
    fs.ensureDirSync(buildFolderPath);

    for (let contractFile in output.contracts) {
        for (let key in output.contracts[contractFile]) {
            fs.outputJsonSync(
                path.resolve(buildFolderPath, `${key}.json`),
                {
                    abi: output.contracts[contractFile][key]["abi"],
                    bytecode: output.contracts[contractFile][key]["evm"]["bytecode"]["object"]
                },
                {
                    spaces: 2,
                    EOL: "\n"
                }
            );
        }
    }
    console.log('Build finished successfully!\n');
} else {
    console.log('\nBuild failed\n');
}
```

***Fuente*: https://medium.com/coinmonks/compiling-deploying-and-interacting-with-smart-contract-using-javascript-641cf0342824**

- **Java**

Java es un lenguaje de programación de contratos inteligentes que es popular y muy solicitado. Es un lenguaje de programación orientado a objetos y basado en clases que fue creado por *Sun Microsystems* en 1995. Gran parte de su sintaxis y estructura se ha derivado de C ++. Java se puede utilizar para crear contratos inteligentes en NEO.

Java fue diseñado para ofrecer flexibilidad a los desarrolladores para escribir código que se ejecutará en cualquier máquina, independientemente de la plataforma o arquitectura. El lenguaje de programación Java se utiliza para crear contratos inteligentes en la cadena de bloques NEO.

Imagen 9: Ejemplo Java

```
pragma solidity >=0.4.22 <0.7.0;

contract PassportService {
    address constant  ADMIN_ADDRESS = 0x90F8bf6A479f320ead074411a4B0e7944Ea8c9C1;
    mapping(address => string) private passportDictionary;
    HistoryRecord[] private historyRecords;

    function createOrUpdatePassport(address _owner, string memory _data) public {
        checkAdminPermission();
        passportDictionary[_owner] = _data;
        historyRecords.push(HistoryRecord({incidentTime : now, owner : _owner, data : _data}));
    }

    function getPassport() public view returns (string memory) {
        return passportDictionary[msg.sender];
    }

    function getHistoryRecord(uint index) public view returns
    (uint256 incidentTime, address owner, string memory data) {
        checkAdminPermission();
        return (historyRecords[index].incidentTime,
                historyRecords[index].owner, historyRecords[index].data);
    }

    struct HistoryRecord {
        uint256 incidentTime;
        address owner;
        string data;
    }

    // utility functions
    function getHistoryRecordLength() public view returns (uint) {
        checkAdminPermission();
        return historyRecords.length;
    }

    function checkAdminPermission() private view {
        if (msg.sender != ADMIN_ADDRESS) {
            revert();
        }
    }
}
```

***Fuente*: https://dzone.com/articles/blockchain-simplified-with-ethereum-example-with-j**

- **Golang**

Go o Golang es un lenguaje de programación de código abierto desarrollado por Google. Soporta programación concurrente, lo que significa que permitirá que múltiples procesos se ejecuten simultáneamente. Se basa libremente en la sintaxis del lenguaje de programación C.

Es un lenguaje sencillo para los desarrolladores. El número de desarrolladores de Golang en todo el mundo se estima en 800.000. Una gran parte del código de cadena de Hyperledger creado con Hyperedger Fabric para contratos inteligentes está escrito en el lenguaje de programación Golang.

Imagen 10: Ejemplo Golang

```
func main() {
        client, err := ethclient.Dial("http://127.0.0.1:7545")
        if err != nil {
                panic(err)
        }

        privateKey, err := crypto.HexToECDSA("PRIVATE_KEY")
        if err != nil {
                panic(err)
        }

        publicKey := privateKey.Public()
        publicKeyECDSA, ok := publicKey.(*ecdsa.PublicKey)
        if !ok {
                panic("invalid key")
        }

        fromAddress := crypto.PubkeyToAddress(*publicKeyECDSA)
        nonce, err := client.PendingNonceAt(context.Background(), fromAddress)
        if err != nil {
                panic(err)
        }

        chainID, err := client.ChainID(context.Background())
        if err != nil {
                panic(err)
        }

        auth, err := bind.NewKeyedTransactorWithChainID(privateKey, chainID)
        if err != nil {
                panic(err)
        }
        auth.Nonce = big.NewInt(int64(nonce))
        auth.Value = big.NewInt(0)      // in wei
        auth.GasLimit = uint64(3000000) // in units
        auth.GasPrice = big.NewInt(1000000)

        auth, err := config.Auth(client)
        if err != nil {
                panic(err)
        }

        address, tx, instance, err := api.DeployApi(auth, client)
        if err != nil {
                panic(err)
        }

        fmt.Println(address.Hex())

        _, _ = instance, tx
}
```

***Fuente*: https://towardsdev.com/creating-a-simple-ethereum-smart-contract-in-golang-138b9439f64e**

- **Vyper**

Vyper es un lenguaje de programación pitónico orientado a contratos que se dirige a la máquina virtual Ethereum (EVM). Dentro de las principales características de Vyper de este lenguaje es la seguridad, la simplicidad del compilador y el lenguaje y la auditabilidad al ser legible por humanos de forma simple.

Imagen 11: Ejemplo Vyper

```vyper
# Open Auction

# Auction params
# Beneficiary receives money from the highest bidder
beneficiary: public(address)
auctionStart: public(uint256)
auctionEnd: public(uint256)

# Current state of auction
highestBidder: public(address)
highestBid: public(uint256)

# Set to true at the end, disallows any change
ended: public(bool)

# Keep track of refunded bids so we can follow the withdraw pattern
pendingReturns: public(HashMap[address, uint256])

# Create a simple auction with `_bidding_time`
# seconds bidding time on behalf of the
# beneficiary address `_beneficiary`.
@external
def __init__(_beneficiary: address, _bidding_time: uint256):
    self.beneficiary = _beneficiary
    self.auctionStart = block.timestamp
    self.auctionEnd = self.auctionStart + _bidding_time

# Bid on the auction with the value sent
# together with this transaction.
# The value will only be refunded if the
# auction is not won.
@external
@payable
def bid():
    # Check if bidding period is over.
    assert block.timestamp < self.auctionEnd
    # Check if bid is high enough
    assert msg.value > self.highestBid
    # Track the refund for the previous high bidder
    self.pendingReturns[self.highestBidder] += self.highestBid
    # Track new high bid
    self.highestBidder = msg.sender
    self.highestBid = msg.value

# Withdraw a previously refunded bid. The withdraw pattern is
# used here to avoid a security issue. If refunds were directly
# sent as part of bid(), a malicious bidding contract could block
# those refunds and thus block new higher bids from coming in.
@external
def withdraw():
    pending_amount: uint256 = self.pendingReturns[msg.sender]
    self.pendingReturns[msg.sender] = 0
    send(msg.sender, pending_amount)

# End the auction and send the highest bid
# to the beneficiary.
@external
def endAuction():
    # It is a good guideline to structure functions that interact
    # with other contracts (i.e. they call functions or send Ether)
    # into three phases:
    # 1. checking conditions
    # 2. performing actions (potentially changing conditions)
    # 3. interacting with other contracts
    # If these phases are mixed up, the other contract could call
    # back into the current contract and modify the state or cause
    # effects (ether payout) to be performed multiple times.
    # If functions called internally include interaction with external
    # contracts, they also have to be considered interaction with
    # external contracts.

    # 1. Conditions
    # Check if auction endtime has been reached
    assert block.timestamp >= self.auctionEnd
    # Check if this function has already been called
    assert not self.ended

    # 2. Effects
    self.ended = True

    # 3. Interaction
    send(self.beneficiary, self.highestBid)
```

***Fuente*: https://vyper.readthedocs.io/en/v0.2.12/vyper-by-example.html**

3.1.7.3. Vulnerabilidades

Aunque los contratos inteligentes parecen fáciles de implementar y comprender, casi nunca es el caso en situaciones del mundo real, a medida que avanzamos hacia contratos inteligentes más escalables.

(Parizi y col, 2021) [52] realizó una evaluación empírica de los lenguajes de Solidity, Pact y Liquidity basada en la usabilidad y seguridad para el desarrollo de nuevos contratos inteligentes.

Los resultados de sus experimentos sugirieron que, en términos de usabilidad, Solidity es el lenguaje más utilizable para que un nuevo desarrollador escriba contratos inteligentes, pero, cuando llega a la seguridad, los nuevos desarrolladores tienden a escribir contratos vulnerables que pueden ser utilizados por entidades malintencionadas para causar daños económicos. [52]

Uno de esos infames ataques maliciosos tuvo lugar en junio 2016, cuando el contrato inteligente DAO (Organización Autónoma Descentralizada) fue manipulado para robar alrededor de 2 millones (50 millones de dólares) de éter. [52]

En otro trabajo, Nicola Atzei, PhD, de *University of Cagliari*. proporcionó un resumen y un breve análisis de algunas vulnerabilidades de seguridad de Solidity y la plataforma Ethereum. Introdujeron y clasificaron una taxonomía de causas de vulnerabilidades en tres niveles: Solidity, EVM (Ethereum *Virtual Machine*) y *Blockchain*. Además, los autores acompañaron esta taxonomía con datos de ataques que aprovechan estas vulnerabilidades. [52]

Los estudios e incidentes mencionados anteriormente muestran que tanto los desarrolladores inexpertos como los experimentados, a menudo pueden escribir contratos inteligentes vulnerables, que pueden ser propensos a fallas las cuales pueden ser explotadas por entidades maliciosas. [50]

Reconociendo este desafío de redactar contratos inteligentes seguros y protegidos, investigadores de comunidades académicas e industriales han comenzado recientemente a centrar su atención sobre el uso de métodos formales para la verificación de contratos inteligentes antes de que se implementen en el *Blockchain*. El proceso de verificación formal implica probar que un código de contrato es correcto para todos los insumos en su espacio

de estados y, por tanto, verificar que el contrato se comporta de acuerdo con su especificación.

3.1.8. Tipos de redes Blockchain

Las redes *Blockchain* pueden clasificarse en tres categorías: por la definición de la participación de los actores y cómo se obtiene esa capacidad de participación, obteniendo redes públicas, privadas e híbridas [53] [54].

También se clasifican en función de quien pueda escribir sobre la red, de esta manera obtenemos redes con permisos (*permissioned network*) o redes sin permisos (*permissionless network*). De esta manera, es posible generar diversas combinaciones en relación a la participación y el permiso sobre la red con sus propias características tal como se muestra en la siguiente tabla [53]:

Públicas		
	Con permiso	**Sin permiso**
Definición	Algunos participantes de la red poseen mayores permisos en relación a otros	Todos los participantes de la red poseen igualdad de condiciones
Objetivo	Centralización	Escalabilidad
Debilidad	Privacidad	Privacidad
Ejemplos de red	EOS, Ripple	Bitcoin, Ethereum

Tabla 5: Tipos de redes *Blockchain* Públicas

Privadas		
	Con permiso	**Sin permiso**
Definición	Se asemeja a una red corporativa privada vista desde la definición de *Blockchain*	Colaboración entre empresas en las que necesariamente exista confianza de tal manera que implemente un consorcio
Objetivo	Centralización	Escalabilidad
Debilidad	Consenso	Consenso
Ejemplos de red	Hyperledger, CORDA	Holochain

Tabla 6: Tipos de redes *Blockchain* Privadas

3.1.8.1. Redes Privadas

Se denominan este tipo de redes a aquellas en la que solo se le permiten añadir y verificar nuevos bloques a determinados usuarios autorizados por la entidad que controla la red (llámese autoridad), que puede ser centralizada o descentralizada [53] [54] [55].

Existe mucho debate acerca del enfoque de este tipo, que choca con las creencias e ideales de conocedores del tema; sin embargo, la aplicación de la tecnología es válida, aunque pueda que no tenga los mismos principios o asegure la confiabilidad que la publica brinda [43].

En este tipo de redes los usuarios no autorizados no pueden añadir información a la cadena de bloques e incluso tener prohibido el acceso a lectura, ya que la autoridad tiene la capacidad de personalizar los accesos de lectura y escritura por individual a cada participante dentro de la red [53].

La trazabilidad de la información en la red puede ser almacenada de forma distribuida, siendo más resilientes y garantizando la replicación de la información en los nodos [53].

En relación al consenso, el algoritmo asociado no requiere un coste computacional o de recursos alto ya que todos los usuarios están autorizados por la autoridad creando un escenario de total o parcial confianza entre los participantes, lo que conlleva a una alta velocidad de transacción [53] [54].

En relación a las desventajas de este tipo de red podemos mencionar [54]:

- Puntos de fallo: dado que son redes que no están descentralizadas en su totalidad, los puntos de fallo y objetivo vulnerable son mayores.
- Es necesario que los participantes confíen plenamente en los nodos validadores de las transacciones, ya que en un caso de que estos resulten comprometidos, podrían manejar de forma fraudulenta la red.

Algunos casos de uso principales para este tipo de red pueden ser [53]:

- Varias organizaciones que deseen trabajar juntas dentro de una red *Blockchain*, pero sin existencia de confianza entre ellas. La red proporcionaría esa confianza en menor o mayor medida en función del tipo de algoritmo de consenso utilizado.

- Una organización desea tener el control absoluto de la red y de los datos almacenados en ella, además de restringir el acceso a dichos datos. Esto implica una total confianza de los participantes en la organización.
- Cualquier escenario aplicable en el que no exista confianza total o parcial entre las partes, la probabilidad de censura sea inexistente o casi nula y que no sea importante que la red y los datos sean expuestos al público.

3.1.8.2. Redes Públicas

La principal característica de este tipo de redes es que cualquiera puede convertirse en usuario de la red, arrancar un nodo que verifique y añada bloques a la red. En la práctica, se deben tener en cuenta una serie de variables que dependen de la red utilizada [53] [54] [55].

Un ejemplo podría ser, una red que implemente el algoritmo de consenso de *Proof of Work* en la cuál para añadir nuevos bloques se debe resolver de forma computacional un problema matemático (por definición e implementación del algoritmo) que implica un gasto excesivo de nivel computacional y consumo de recursos [53].

En la práctica, no cualquier usuario podría añadir nuevos bloques a este tipo de redes. Esta restricción se considera necesaria, ya que, al no existir, cualquier usuario podría tener intenciones maliciosas para la red como la propia sobrecarga el sistema o añadir bloques incorrectos con el fin de obtener un beneficio, por mencionar algunos casos [53].

Las medidas para evitar este tipo de intenciones maliciosas varían dependiendo del algoritmo de consenso. Por otro lado, se incentivan a usuarios de la red a ser “creadores/validadores de bloques” (mineros), obteniendo una recompensa cada vez que se añade un nuevo bloque [53].

Las principales ventajas de este tipo de red son [54]:

- Son cadenas totalmente descentralizadas por lo que la información se distribuye a través de todos los nodos que conforman la red.
- Se mantiene en todo momento el anonimato de los participantes.
- La transparencia, todos los participantes intervienen de igual forma en la cadena de bloques y tienen accesible la misma información.

- Dado que la información está expuesta en público, se permite la realización de transacciones en un entorno no seguro.

Como desventajas de este tipo de red podemos mencionar las siguientes [54]:

- Requiere gran consumo de recursos en función del algoritmo de consenso utilizado.
- Existe un número de transacciones limitado que se pueden introducir dentro del bloque.
- El anonimato puede considerarse una desventaja en ámbitos en los que existen requerimientos más estrictos sobre la identidad y seguridad.

3.1.8.3. Redes Híbridas, de Consorcio o Federadas

Se definen como una combinación de los dos tipos de redes anteriores. La red no está controlada por la comunidad como el caso de las redes públicas ni por una entidad autoritaria como el caso de las redes privadas, sino que es controlada por un conjunto de nodos determinados (consorcio) [53] [43].

El acceso a la información de la red podría estar restringido a determinados usuarios o ser público. Se considera que están parcialmente centralizadas, ya que establece un conjunto de nodos generadores de bloques, determinando cierto grado de centralización o de descentralización dependiendo de la cantidad de nodos con esta función [53].

Al ser una combinación de redes públicas y privadas, pueden aprovechar las virtudes de cada tipo evitando los defectos. Uno de los beneficios destacados es la protección contra el ataque del 51% ya que la incorporación a la red está limitada [53].

La siguiente tabla muestra una comparación de los tipos de redes abordados [53]:

Tipos de redes

Propiedad	Pública	Privada	Híbrida
Productores	Cualquier usuario	Una entidad	Conjunto de nodos seleccionados
Permisos	Públicos	Públicos, parcialmente públicos o restringidos	Públicos, parcialmente públicos o restringidos
Inmutabilidad	Prácticamente garantizada	No asegurada	No asegurada
Centralización	No	Si	Parcial

Tabla 7: Tipos de redes

Bitcoin

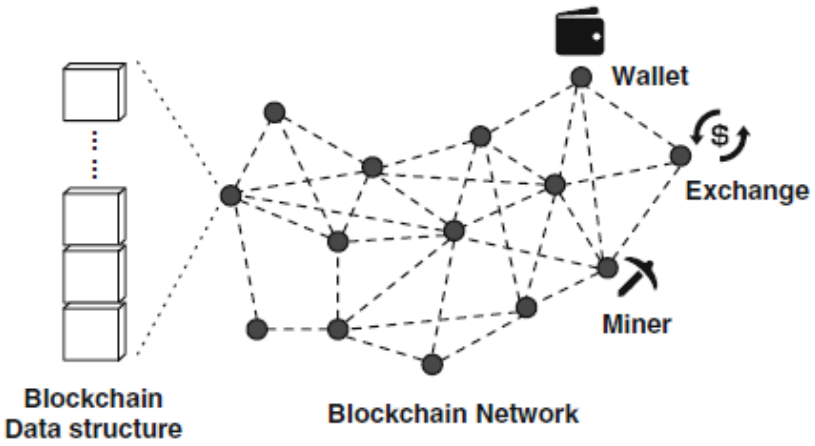
Bitcoin (BTC) es una criptomoneda operada en una red *peer-to-peer*. A diferencia de sistemas bancarios y de pago tradicionales, Bitcoin se basa en la confianza descentralizada; no existe una autoridad central de confianza en el sistema Bitcoin. La confianza surge de las interacciones de diferentes participantes en el ecosistema.

La imagen 11 da una descripción general del sistema Bitcoin. En el sistema Bitcoin, hay un libro mayor distribuido que almacena todas las transacciones de Bitcoin. El contenido del libro mayor se replica en muchos nodos de procesamiento distribuidos geográficamente dentro de la red Bitcoin.

Hay tres tipos principales de nodos dentro de la red Bitcoin:

- **Usuarios con billeteras,** una billetera mantiene los pares de claves del usuario, que se utilizan para autenticar las transacciones iniciadas por el usuario mediante firmas digitales.
- **Los mineros,** que compiten entre sí para agregar nuevos bloques al libro mayor compartido como fuente autorizada de todas las transacciones.
- **Exchange,** es decir, lugares donde los usuarios puede comprar BTC a cambio de otras monedas.

Imagen 12: Ejemplo Bitcoin



***Fuente*: X. Xu, I. Weber and M. Staples, Architecture for block chain applications, 1st ed. Switzerland: Springer Nature, 2019 Página 28**

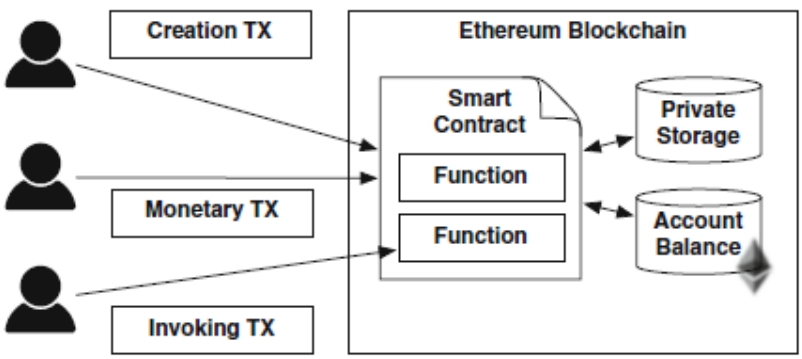
Ethereum

Bitcoin lideró el desarrollo de la primera generación de sistemas *Blockchain*, proporcionando un libro mayor público para registrar transacciones financieras firmadas criptográficamente.

Bitcoin tiene soporte limitado para transacciones programables, y solo piezas muy pequeñas de datos auxiliares pueden incorporarse a las transacciones para otros fines.

La plataforma de código abierto Ethereum, dentro de la segunda generación de sistemas *Blockchain*, proporciona una infraestructura programable de propósito general donde el libro público no solo almacena transacciones financieras, sino también tiene las facilidades para implementar y ejecutar programas en el sistema *Blockchain*. La plataforma *Blockchain* de Ethereum ve el contrato inteligente como un elemento de primera clase e incluye una máquina virtual para ejecutar contratos inteligentes.

Imagen 13: Ejemplo de Ethereum



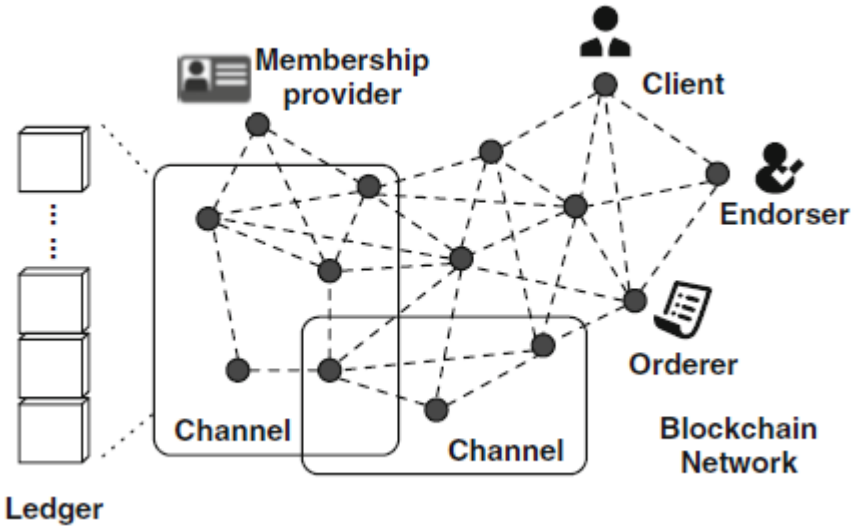
***Fuente*: X. Xu, I. Weber and M. Staples, Architecture for block chain applications, 1st ed. Switzerland: Springer Nature, 2019 Página 38**

Hyperledger

Hyperledger es un proyecto general de cadenas de bloques de código abierto y herramientas relacionadas. Es una colaboración global, organizada por la Fundación Linux desde diciembre 2015. Los miembros son de dominios como finanzas, banca, Internet de las cosas, cadenas de suministro, fabricación y tecnología. Actualmente hay más de 185 miembros y 8 proyectos en curso, incluido *Hyperledger Fabric.*

Hyperledger Fabric es un marco de *Blockchain* empresarial, destinado a ser una base para el desarrollo de aplicaciones basadas en *Blockchain* con una arquitectura modular. Los datos se pueden almacenar en múltiples formatos y varios algoritmos de consenso pueden ser configurados.

Imagen 14: Ejemplo Hyperledger



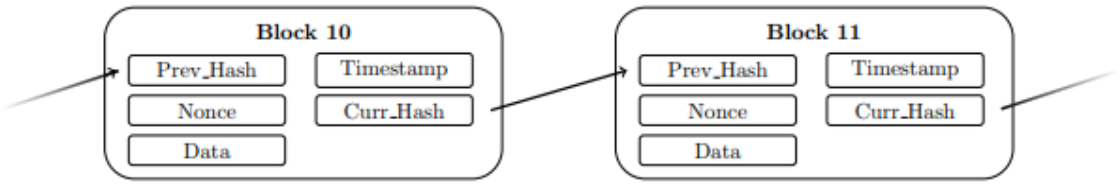
***Fuente*: X. Xu, I. Weber and M. Staples, Architecture for block chain applications, 1st ed. Switzerland: Springer Nature, 2019 Página 40**

3.1.9. Funcionamiento de una red Blockchain

Un bloque es un conjunto de transacciones confirmadas e información adicional que será incluida dentro de la cadena de bloques. Cada bloque que forma parte de la cadena (exceptuando el bloque génesis o bloque generatriz) contiene al menos la siguiente información [56] [43].

- Un número de bloque.
- Un código alfanumérico o *hash* que enlaza con el bloque previo.
- Un paquete de transacciones, o en general, la información (data, *payload* o carga útil).
- Un código alfanumérico o *hash* propio del bloque, que será el punto de enlace o referencia con el siguiente bloque.
- Un número aleatorio llamado *nonce* (*number that can be only used once*) o número que solo puede usarse una vez.
- Una marca de tiempo que registra el momento de la creación del bloque.

Imagen 15: Funcionamiento de una red *Blockchain*

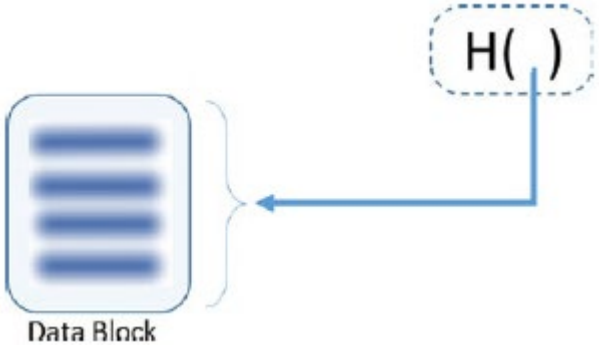


***Fuente*: A. Shanti Bruyn, "Blockchain an Introduction", University Amsterdan, pp. 19-40, 2017.**

El código alfanumérico propio y único de cada bloque está relacionado a un puntero *hash*. Un puntero *hash* es un *hash* criptográfico, comúnmente obtenido con funciones de las familias de *hash SHA2* o *SHA3*, que identifica un bloque de datos de forma particular, donde el puntero *hash* es el resultado de una función *hash* de la información contenida en el bloque de datos en sí (encabezado más data).

A diferencia de las listas enlazadas que apuntan al siguiente bloque para que se pueda acceder a él, los punteros *hash* se encargan de apuntar al bloque de datos anterior y así proporcionar una forma de verificar que los datos no han sido manipulados [56].

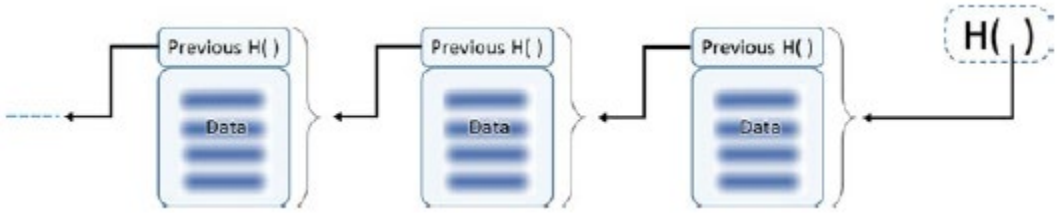
Imagen 16: Ejemplo de la data de un bloque.



***Fuente*: B. Singhal, G. Dhameja and P. Panda, "How Blockchain Works", Beginning Blockchain, pp. 31-148, 2018.**

El propósito del puntero *hash* es construir una red de bloques resistente a la manipulación que puede considerarse como una única fuente de confianza. El *hash* del bloque anterior se almacena en el encabezado del bloque actual, y el *hash* del bloque actual con su encabezado de bloque, se almacenará en el siguiente encabezado de bloque y así sucesivamente. Esto permite crear una cadena de bloques tal como se muestra en la siguiente figura [56].

Imagen 17: Ejemplo de una cadena de bloques

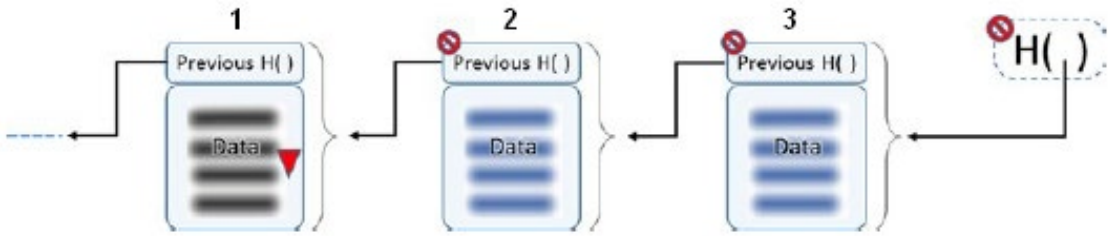


***Fuente*: B. Singhal, G. Dhameja and P. Panda, "How Blockchain Works", Beginning Blockchain, pp. 31-148, 2018.**

Como podemos observar, cada bloque apunta a su bloque anterior, conocido como "El bloque padre". Cada nuevo bloque que se agrega a la cadena se convierte en el bloque padre para el siguiente bloque a ser agregado en la cadena. Esto es así hasta llegar al primer bloque que se crea en la cadena de bloques, que se denomina "bloque generador o bloque génesis" [56].

En un diseño de este tipo, en el que los bloques se vinculan con *hash*es, prácticamente no es factible que se alteren los datos en cualquier bloque sin tener una afectación e impacto en todo el resto de la cadena. Dado que una de las propiedades de las funciones *hash* consiste en que, si se altera el valor de la data de entrada para el cálculo, el valor del *hash* resultante también cambia, por lo que no habría coincidencia con los *hash*es que ya han sido registrados [56].

Imagen 18: Ejemplo bloques vinculados con hashes



***Fuente*: B. Singhal, G. Dhameja and P. Panda, "How Blockchain Works", Beginning Blockchain, pp. 31-148, 2018.**

Para ejemplificar la imagen anterior, se tienen los siguientes casos:

- Cualquier intento de cambiar el encabezado o el contenido del bloque rompe toda la cadena. Suponiendo que se modifican los datos del bloque 1, el *hash* que se almacena en el encabezado del bloque de bloque 2 no coincidiría [56].
- Si se llegase a cambiar el *hash* almacenado en el encabezado del bloque 2 para que coincida con el nuevo *hash* con data alterada del bloque 1, el *hash* del bloque 2 cambia (ya que el *hash* representa al encabezado y los datos del bloque 2) y no tendrá coincidencia con el almacenado en el encabezado de bloque del bloque 2 [56].
- Para mantener la integridad de la cadena de bloques, se volvería necesario repetir este proceso hasta llegar al *hash* más reciente. Dado que todos o muchos nodos en la red ya tienen una copia de la cadena de bloques con el *hash* más reciente, de ninguna manera es posible piratear la mayoría de los sistemas y cambiar todos los *hash*es a la vez [56].
- Esto lo convierte en una estructura de datos de cadena de bloques a prueba de manipulaciones.

A nivel de profundidad en el caso de las redes *Blockchain* públicas, es necesario mencionar el trabajo de los mineros o *miners*. Los mineros son ordenadores o equipos especializados dedicados que aportan poder computacional a la red. Esta potencia se usa para verificar las transacciones que se llevan a cabo. Cada vez que un nodo minero completa un bloque, recibe una recompensa por cada bloque “encontrado”. A continuación, se explica el proceso seguido por este tipo de rol dentro de la red [57] [43].

1. Cada nueva transacción es enviada y compartida a todos los nodos de la red *Blockchain* (incluyendo los mineros) [57].
2. Cada nodo minero colecta nuevas transacciones dentro de la red [57].

3. Cada nodo trabaja en “encontrar” un nuevo bloque mediante el algoritmo de consenso y cálculo de *hash*es criptográficos tomando como base el *hash* del bloque anterior, la marca de tiempo del bloque, la raíz Merkle del bloque y la dificultad de la red, los cuáles se explican a detalle más adelante [57] [59].
4. Cuando un nodo minero encuentra un *hash* válido, este envía y comparte el bloque a todos los nodos restantes incluyendo el *nonce* utilizado [57] [59].
5. Los nodos aceptan el bloque únicamente si todas las transacciones contenidas en el bloque son válidas, verificando además si el *hash* resultante corresponde realmente con el valor del *nonce* compartido en el paso 4, lo cual es una tarea que se lleva a cabo en menor tiempo en comparación a la búsqueda del *hash* realizada por los mineros [57] [59].
6. Los nodos expresan la aceptación del bloque iniciando el proceso de creación del siguiente bloque de la red tomando como *hash* previo el valor del *hash* del bloque aceptado en el paso 5 [57].

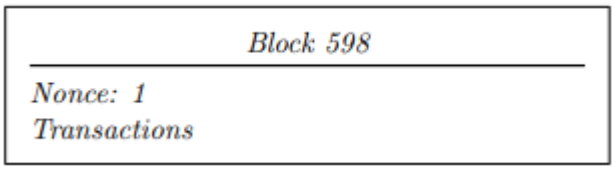
El cálculo del *hash* es el problema matemático de complejidad alta que los mineros deben resolver para encontrar o añadir un nuevo bloque a la red. Esto es posible mediante la implementación del protocolo de consenso seleccionado por la red. El algoritmo comúnmente utilizado es el de “prueba de trabajo” (*PoW*, por sus siglás en inglés *Proof of Work*) el cuál funciona como sigue [57].

- Se genera un número aleatorio, llamado *nonce* que es utilizado para imponer un determinado nivel de cómputo en las inserciones de bloques dentro del *Blockchain*; simultáneamente, es utilizado para satisfacer ciertas condiciones propias de cada red; este valor puede ser generado ya sea una vez o una cantidad indeterminada de veces, hasta que el valor *hash* del bloque, coincida con las restricciones mencionadas anteriormente [57] [el de Guatemala].
- El valor *nonce* calculado es agregado a la sección de data del bloque [57].
- Se calcula el valor del *hash* como resultado de una función criptográfica (como por ejemplo SHA256) [57].
- La dificultad del cálculo del *hash* para los mineros corresponde con encontrar un valor de *hash* que cumpla con la condición definida dentro de la red, por ejemplo, se considerarán *hash*es válidos todos aquellos cuyos “n” primeros valores iniciales sean ceros [57].
- Si el *hash* del bloque calculado inicia con un número determinado de ceros (de acuerdo a la dificultad), entonces un nuevo bloque es “encontrado” y agregado a la red. Caso contrario, los mineros deben iniciar el proceso nuevamente en el paso 1 generando un nuevo valor para *nonce* [57].

El proceso anterior se ejemplifica a continuación [57]:

El nodo minero A inicia el proceso de búsqueda de un *hash* aceptable para un nuevo bloque 598, siendo la dificultad de la red todo *hash* que inicie con 4 ceros [57].

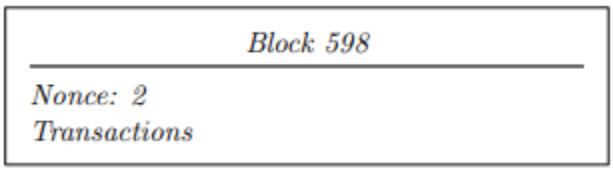
Imagen 19: Ejemplo nodo minero



***Fuente*: A. Shanti Bruyn, "Blockchain an Introduction", University Amsterdan, pp. 19-40, 2017.**

El *hash* correspondiente a la configuración actual del bloque uno es: bac6d67daf63c7a06bab569adeadab130d332ed4c870da314d87f6f1b4c8a409. De acuerdo a la dificultad impuesta, este valor de *hash* no es aceptable y se reinicia el proceso con un nuevo valor de *nonce* [57].

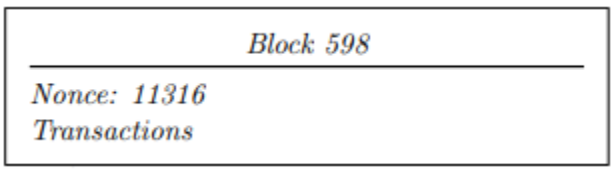
Imagen 20: Ejemplo de reinicio del proceso



***Fuente*: A. Shanti Bruyn, "Blockchain an Introduction", University Amsterdan, pp. 19-40, 2017.**

El *hash* resultante para la configuración anterior corresponde con 6ee6b0c4aa8e6aaa369dfaee1379a59cec8797e7fc8ad1a358d57e1a87a1466d el cuál no cumple con las restricciones de la red. El nodo minero continúa realizando este proceso de prueba y error hasta encontrar un *hash* que cumpla con el requerimiento de la red. Este *hash* es encontrado hasta que el valor de *nonce* corresponda con 11316 [2].

Imagen 21: Valor encontrado nonce



***Fuente*: A. Shanti Bruyn, "Blockchain an Introduction", University Amsterdan, pp. 19-40, 2017.**

El *hash* resultante de la configuración del bloque anterior es 000015783b764259d382017d91a36d206d0600e2cbb3567748f46a33fe9297cf el cuál

satisface la condición impuesta por la red, por lo que el bloque es “encontrado” y añadido a la red [57].

Este proceso descrito es el más “costoso” ya que requiere un poder computacional y energía para poder resolverlo [57].

El *nonce* también resulta ser un valor de gran importancia en otras redes *Blockchain*. Por ejemplo, en otras redes públicas y aquellas orientadas a criptomonedas, tienen en su programación la generación y uso de este tipo de número. De hecho, cualquier sistema basado en *Proof of Work* (*PoW*) usa *nonce* para realizar este trabajo [59].

Sin embargo, las *Blockchain* con protocolo de consenso *PoW* no son las únicas en usar *nonce*. Las *Blockchain* que usan el protocolo de Prueba de Participación (*PoS* por sus siglas en inglés *Proof of Stake*) también usan el *nonce* [59].

Lo mismo podemos decir de las redes *Blockchain* privadas que usan la Prueba de Autoridad (*PoA* por sus siglas en inglés *Proof of Authority*) o la Prueba de Tiempo Transcurrido (*PoET* por sus siglas en inglés *Proof of Elapsed Time*), esta última ligada al desarrollo de *Hyperledger*. El desarrollo de *Hyperledger* basa su funcionamiento en la creación de *nonce* usando sistemas de generación de números aleatorios habilitados por hardware para alcanzar un mayor rendimiento [59].

Por supuesto, en protocolos como *PoS, PoA y PoET*, el *nonce* es usado con otros objetivos y la dificultad de su cálculo es mucho menor. Pero pese a esto, el concepto aplicado es el mismo, un número aleatorio creado para garantizar la seguridad criptográfica de la *Blockchain* [59].

Árbol Merkle

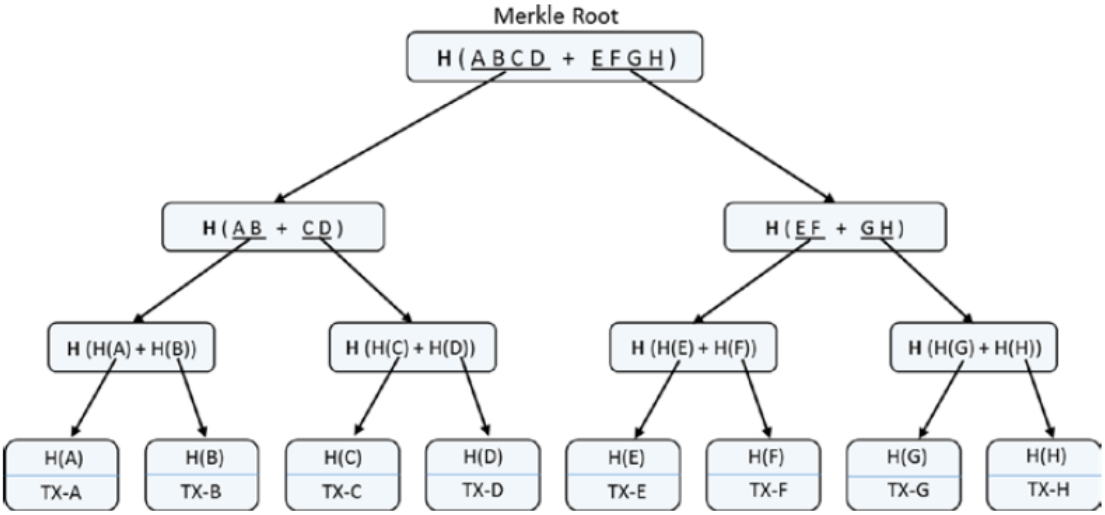
Un árbol Merkle es un árbol binario de punteros *hash* criptográficos, de ahí que es un árbol *hash* binario. Lleva el nombre de su inventor Ralph Merkle. Es otra estructura de datos útil que se utiliza en soluciones *Blockchain* como Bitcoin.

Los árboles de Merkle se construyen mediante el *hash* de datos emparejados (generalmente transacciones en el nivel de hoja), luego se calcula de nuevo el *hash* tomando como entradas las salidas *hashes* calculados previamente hasta llegar el nodo raíz, llamado raíz de Merkle.

Como cualquier otro árbol, es construido de abajo hacia arriba. En las redes *Blockchain* públicas, las hojas son siempre transacciones de un bloque único en una cadena de bloques.

Un ejemplo de árbol de Merkle se muestra a continuación:

Imagen 22: Ejemplo árbol de Merkle



***Fuente*: B. Singhal, G. Dhameja and P. Panda, "How Blockchain Works", Beginning Blockchain, pp. 31-148, 2018.**

Similar a la estructura de datos del puntero *hash*, el árbol Merkle también es a prueba de manipulaciones. La manipulación en cualquier nivel del árbol no coincidiría con el *hash* almacenado en un nivel superior en la jerarquía y hasta el nodo raíz.

Es realmente difícil para un atacante cambiar todos los valores *hash* de todo el árbol. También asegura la integridad del orden de las transacciones. Si cambias únicamente el orden de las transacciones, también los *hashes* en el árbol y por consecuencia la raíz Merkle cambiará.

El árbol Merkle es un árbol binario y debería existir un número par de elementos a nivel de hoja. En caso de que no exista un número par de elementos al nivel de hojas, una solución es duplicar la última transacción y de esa forma, es posible equilibrar el árbol.

Para validar si una transacción ha ocurrido, o encontrar una transacción a partir de su *hash*, la única forma de verificarlo sería navegar por un árbol de Merkle hasta que se encuentre el bloque exacto que coincide con el *hash* de la transacción.

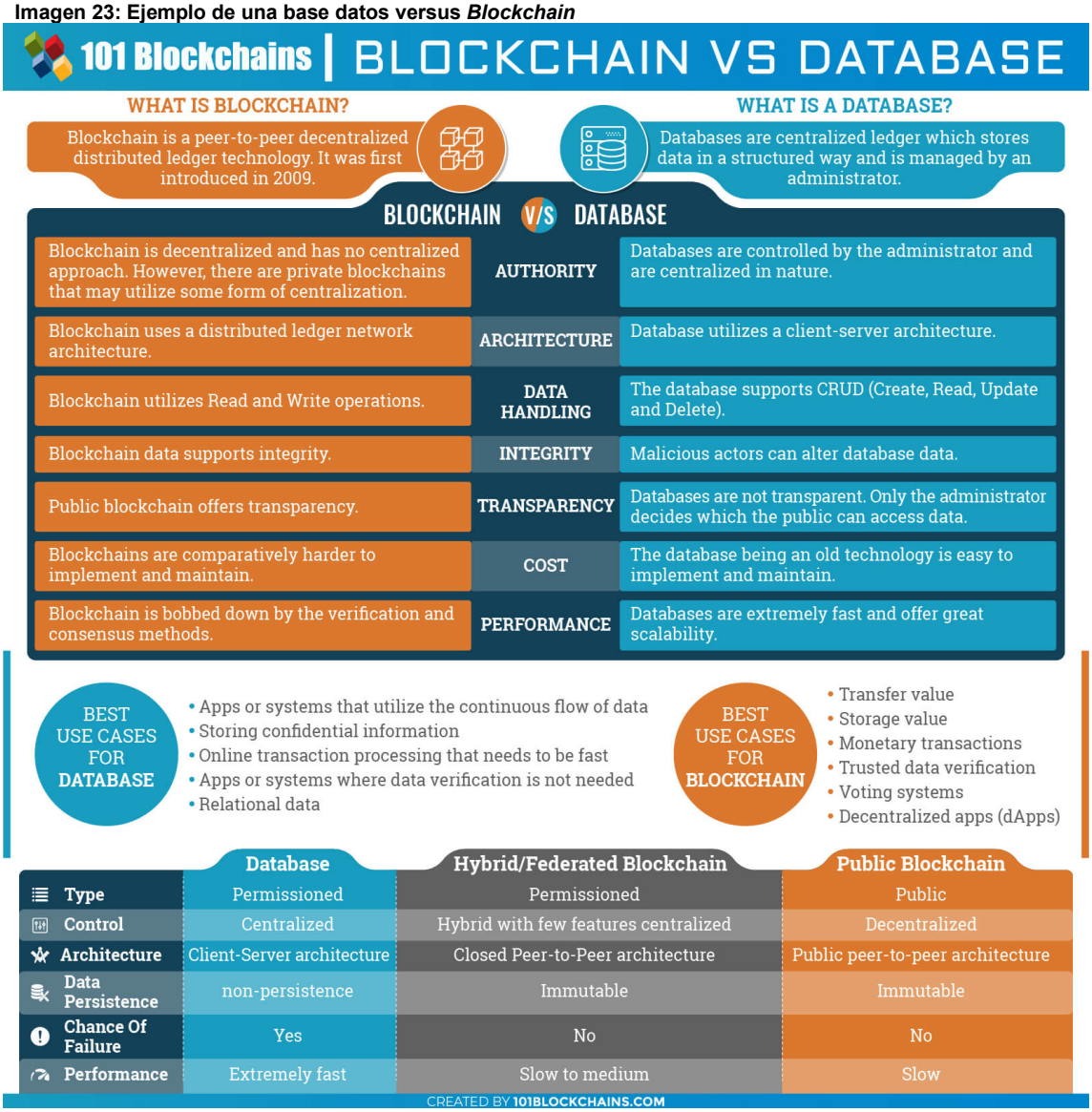
3.1.10. Propiedades no funcionales de una red *Blockchain*

En los capítulos anteriores se ha detallado las características de una red *Blockchain*, los tipos que existen y sus ventajas.

Se puede decir que las propiedades de inmutabilidad, no repudio, integridad, transparencia e igualdad de derechos, son las principales propiedades de una cadena de bloques. Pero existen otras propiedades que es necesario analizar antes de decidir implementar una red *Blockchain*.

Vamos a comparar una Base de Datos tradicional versus una red *Blockchain*, para ello nos apoyaremos del siguiente esquema elaborado por *101Blockchain.com* [60]

Imagen 23: Ejemplo de una base datos versus *Blockchain*



***Fuente*: https://101blockchains.com/blockchain-vs-database-the-difference/**

Con base al esquema profundizaremos en las siguientes propiedades:

- Autoridad y Control
- Arquitectura
- Manejo de Datos
- Transparencia
- Costos y Escalabilidad
- Rendimiento

Autoridad y Control [60]

Lo que hace innovadora la tecnología *Blockchain*, es que no requiere de ninguna entidad de control para funcionar, en cambio una Base de Datos tradicional requiere un administrador quien define las reglas y tiene el poder de modificar la base de datos sin necesidad de realizar un consenso con los usuarios, a esto se le conoce como red centralizada.

En una red centralizada el administrador también puede delegar funciones a otros usuarios que según el rango tiene cierto poder sobre la manipulación de la información, crear respaldos y procesos de mantenimiento.

Los participantes de una red *Blockchain*, confían en la red en sí misma, en lugar de depender de organizaciones de terceros para aprobar sus transacciones. Pero esto, no es tan sencillo cuando existen datos y procesos privados en juego.

Por esta razón *Blockchain* evoluciono, por lo que además de tener una *Blockchain* completamente publica, crearon la cadena de bloques híbrida / privada, donde algunas funciones trabajan de forma centralizada. Este tipo de cadena de bloques, es el que comúnmente resuelve el problema de las organizaciones privadas.

La descentralización de una red *Blockchain* trae muchos cambios de implementación a los sistemas y procesos actuales utilizados por las diferentes industrias.

Arquitectura [60]

En una base de datos tradicional se utiliza la arquitectura cliente / servidor. Aquí el cliente son los receptores, mientras que los servidores actúan como una unidad de procesamiento centralizada. La comunicación entre el cliente y los servidores se mantiene a través de una conexión segura.

En una red *Blockchain* hibrido/privada, la arquitectura que se utiliza es *peer-to-peer* cerrada.

Por otro lado, la red *Blockchain* publica utiliza una arquitectura distribuida, todos los nodos o usuarios de la red se comunican de igual a igual (*peer-to-peer*), utilizando protocolos y herramientas criptográficas que aportan la seguridad a la red. Como no hay un nodo centralizado, los nodos pueden participar colectivamente en el algoritmo de consenso.

Las bases de datos no requieren un algoritmo de consenso y depende completamente del enfoque centralizado. El administrador controla todos los aspectos de la base de datos y está altamente centralizado. Por esta razón el tiempo por transacción es más rápido en comparación con *Blockchain,* pero no proporciona la inmutabilidad de los datos.

Manejo de Datos [60]

Las redes *Blockchain* ya sean hibridas o públicas, garantiza la inmutabilidad de cada transacción, lo que significa que los datos una vez escritos no se pueden borrar ni reemplazar, no es posible la manipulación de datos dentro de la red.

En una base de datos tradicional además de tener funciones de lectura y escritura, los valores ingresados pueden ser eliminados o actualizados, por lo que son vulnerables ante administradores deshonestos o hackeos de terceros.

En resumen, *Blockchain* solo admite dos operaciones, lectura y escritura.

- **Operaciones de lectura:** se utiliza para leer o recuperar datos de la red *Blockchain*.
- **Operaciones de escritura:** se utiliza para agregar información y datos a la red *Blockchain*.

Transparencia [60]

La transparencia es una de las propiedades más sobresalientes de la tecnología *Blockchain*, también es la más criticada. En una *Blockchain* de tipo público, la privacidad es limitada, no hay usuarios con súper privilegios, cualquier usuario puede verificar las transacciones una vez escritas en la red y validar nuevas transacciones.

En cambio, en una red centralizada, no existe ninguna forma de transparencia. Solamente el administrador de la base de datos cuenta con el privilegio de validar las transacciones, incluso puede permitir que un conjunto de datos sea público, pero aun así, una persona no puede realizar la verificación de los datos.

Al considerar implementar una red *Blockchain* en una aplicación comercial, se considera necesario concertar un intercambio aceptable entre la confidencialidad de los datos y la transparencia, por lo que se sugiere implementar una *Blockchain* Hibrida.

Costos y Escalabilidad [60] [61]

A nivel de gastos de implementación, la base de datos tradicional es menos costosa en comparación a implementar una red *Blockchain*.

Actualmente en el mercado existen diferentes bases de datos, listas para ser implementadas a la medida a cualquier tipo de empresa y se adaptan muy bien con todas las tecnologías incluyendo sistemas legados, aportando escalabilidad a la red.

Por otra parte, las cadenas de bloques públicas por ser una tecnología relativamente nueva que sigue evolucionando, tienen límites de escalabilidad, como los siguientes:

- El tamaño de los datos en *Blockchain*, debido a la replicación global de todos los datos en todos los nodos completos.
- La tasa de procesamiento de transacciones. Por ejemplo, cadenas de bloques públicas convencionales solo puede manejar un promedio de 3 a 20 transacciones por segundo, mientras que los principales servicios de pago, como VISA, manejan un promedio de 1700 transacciones por segundo. [60] [61]

- La latencia de la transmisión de datos. Dado que los nodos pueden tener una copia local del *Blockchain*, la latencia de lectura puede ser alta, pero debido a que las actualizaciones deben propagarse en una red global, la latencia de escritura suele ser alta.
- *Blockchain* requiere una implementación de extremo a extremo y no puede integrarse simplemente en un sistema existente como un complemento. Las empresas interesadas en esta tecnología deben realizar una planificación y ejecución adecuadas.

Conociendo ya las limitantes a nivel de costos, también se deben analizar los beneficios a largo plazo, si miramos más a fondo, el costo asociado con la tecnología *Blockchain* podría proporcionar una solución más rentable ya que los pares o nodos son los que administran principalmente la red.

Las organizaciones con red *Blockchain* no tendrían que lidiar con el costo adicional asociado con el manejo de la red, mantenimiento o gastos por fallas o interrupción de servicio, lo que puede ahorrar muchos costos en el presupuesto anual de una empresa.

Rendimiento [60]

La velocidad de transacción en una base de datos tradicional es más rápida que en una red *Blockchain*. Pero se debe considerar que cuando se realiza una transacción en una cadena de bloques, esta realiza todas las operaciones que se llevan a cabo en una base de datos tradicional y además procesos criptográficos.

La verificación de firmas digitales, los mecanismos de consenso y la redundancia son algunas de las operaciones que una red *Blockchain* realiza para asegurar la integridad de cada transacción. Esto requiere de un alto nivel de procesamiento lo que provoca mayor latencia.

Con base a lo anterior podemos decir que la base de datos tradicional sobresale cuando se trata de utilidad, velocidad y precisión. Sin embargo, *Blockchain* también es un ganador cuando se trata de innovación, verificación y automatización por lo que se vuelve la mejor opción si se busca confianza, transparencia y verificación.

3.2. Consideraciones de seguridad

3.2.1. Seguridad en *Blockchain* según el uso

Las características y beneficios de la tecnología *Blockchain* potencializa el comercio global abierto. Sin embargo, para aprovecharlo al máximo, es necesario tomar en cuenta cuáles son sus vulnerabilidades.

La diferencia en la seguridad de una Blockchain publica y privadas pueden tener diferentes implicaciones con respecto a los privilegios de participación y acceso a los datos.

- **Seguridad de red *Blockchain* Publico**

  Las redes Blockchain públicas están abiertas y permiten que cualquier usuario se una a la red y al mismo tiempo garantiza el anonimato de los participantes.

  La cadena de bloques pública aprovecha que las computadoras se encuentran conectadas a la Internet, para validar las transacciones y realizar el mecanismo consenso. [68]

  Para este tipo de redes se recomienda, fortalecer la protección en la distribución y descentralización [68], con el objetivo de disminuir las vulnerabilidades que se detallaran más adelante.

- **Seguridad en red *Blockchain* Privada**

  Las redes privadas de *Blockchain* dependen de la identidad para confirmar la membresía y los privilegios de acceso de cada usuario. Además, permiten la participación únicamente de las organizaciones asociadas. [68]

  Para las redes Blockchain privadas y autorizadas se recomienda implementar controles más estrictos para que todos los miembros cumplan a cabalidad con todas las normas y condiciones de uso de la red según el rango definido para cada usuario.[68]

3.2.2. Principales amenazas de una red Blockchain

Las vulnerabilidades en una cadena de bloques, surgen habitualmente como resultado de la implementación de las plataformas y aplicaciones, es decir, se vinculan al desarrollo del código informático, de los protocolos de comunicación o en los mecanismos de validación y consenso de los bloques.

En este capítulo se describen las principales amenazas que sufre una red Blockchain con énfasis en redes públicas.

Ataque de doble gasto

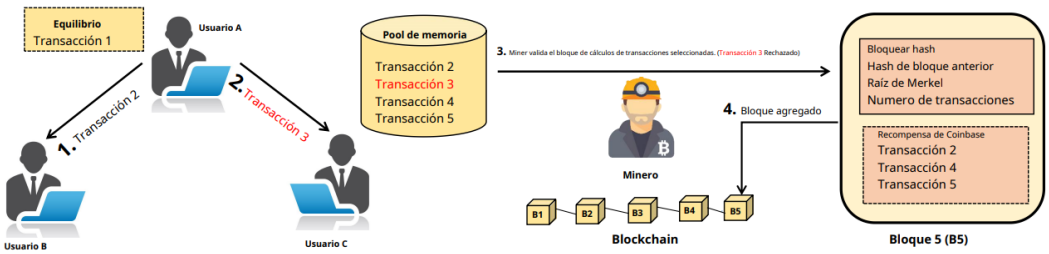
Este tipo de ataques es común en transacciones de Criptomoneda basadas en *PoW*. En Bitcoin el tiempo promedio de minería de bloques es de 10 minutos. En un entorno de transacciones rápidas, la red entrega el producto al remitente antes de que la transacción se mida en *Blockchain*. [62]

El remitente mal intencionado puede utilizar diferentes estrategias o ataques cuya intención es retrasar el tempo de validación del bloque para utilizar en dos veces simultáneamente el mismo token, firmar la misma transacción y enviarla a otro destinatario.

Este comportamiento de firmar la misma transacción con una clave privada y enviarla a dos receptores diferentes se conoce como doble gasto.[62]

En la siguiente figura se plantea un ejemplo de cómo se realiza el ataque de doble gasto

Imagen 24: Ejemplo ataque de doble gasto



***Fuente*: https://arxiv.org/pdf/1904.03487.pdf**

En la imagen 23, se observa que el usuario A tiene la Transacción 1 en su saldo, la usa como entrada para realizar la Transacción 2 con el usuario B y a la vez utiliza el mismo saldo para realizar simultáneamente una Transacción 3 con el usuario C, (cabe mencionar que el usuario A tiene que poseer un gran poder computacional para reescribir la cadena) posteriormente ambas transacciones llegan al Pool de memoria (*mempool*) y el minero puede seleccionar la Transacción 2 o Transacción 3. En esta ilustración el minero selecciona la Transacción 2, es descartada la Transacción 3, por lo tanto, el usuario C sufre la perdida. [62]

Aunque una red *Blockchain* tenga implementado un mecanismo de consenso para validar las transacciones, todavía es imposible al 100% evitar el doble gasto.

Sin embargo, los usuarios receptores de una transacción podrían evitar ser víctimas de este ataque, si logran confirmar entre 3 y 6 bloques posteriores al bloque en el que se incluye su transacción. Esto hace más difícil eliminar el bloque de la cadena, ya que la capacidad computacional necesaria para remplazar no solo ese bloque sino también, los que siguen deberán ser mayor que el resto de toda la red *Blockchain*. [63]

Ataque del 51%

El ataque del 51% también conocido como *The Majority Attack*, es de los más conocidos en las redes *Blockchain*.

Se da cuando la capacidad computacional de la *Blockchain* se concentra en un solo miembro de la red, y su *hash* representa más del 50%, superando la capacidad del resto de usuarios juntos. El atacante que consiga esto, podría añadir nuevos bloques más rápido que los demás y su cadena siempre sería la más larga, es decir que el atacante siempre tendrá ventaja en especial si la red *Blockchain* se basa en *PoW*. [63]

En las cadenas de bloques basadas en *PoS* el ataque del 51% también puede ocurrir, si la cantidad de monedas que posee un solo minero es más del 50% del total de la cadena de bloques. [65]

Al hacer efectivo el ataque, el atacante puede manipular la *Blockchain* de manera arbitraria de la siguiente manera: [65]

- Realizar transacciones con doble gasto

- Excluir y modificar el orden de las transacciones
- Obstaculizar las transacciones mineras normales de otros mineros
- Impedir la operación de confirmación de transacciones normales.

Implicaciones: [62]

- Evitar que las transacciones o bloques sean verificados.
- revertir las transacciones durante el tiempo que tienen el control para permitir el doble gasto.
- bifurcar la *Blockchain* principal y dividir la red.
- prevenir otros mineros (verificadores) para que no encuentren ningún bloque durante un corto período de tiempo.

Contramedidas: Utilizar el algoritmo de consenso *PoW* con doble fase.[63]

Ataque Sybil

Los sistemas *peer-to-peer* a gran escala, sin una autoridad lógicamente centralizada, siempre son vulnerables a los ataques Sybil.[66] Esta afirmación la hace John R. Douceur en su artículo *The Sybil Attack*. La idea de Douceur es sencilla y se puede resumir de la siguiente forma:

Un sistema *P2P* puede ser vulnerado, si buena parte de sus nodos (que se suponen seguros y pertenecen a distintas personas), son en realidad controlados por una misma persona que permanece en las sombras.[67]

El ataque consiste en crear múltiples identidades falsas para conseguir cierta influencia en la red y así poder llevar a cabo acciones contrarias a las normas de la red. [63]

En las redes *Blockchain* se intenta evitar este tipo de ataques mediante los algoritmos de consenso utilizados. En *PoW* se combate mediante la minería de bloques y en el mecanismo *PoS* mediante el sistema pertenecía.[63]

Ya que esta vulnerabilidad es parte del sistema *P2P* son los programadores de la red quienes deben configurar los métodos preventivos. Entre las medidas más usadas para prevenir este tipo de ataque podemos mencionar: [67]

- Usar sistemas de validación y de cadenas de confianza.
- Usar protocolos de consenso que supongan un costo por la creación de una nueva identidad en la red o acceso a recursos de red.
- Creación de un sistema de reputación, que consiste en asignar roles a los diferentes usuarios según el tiempo que han pertenecido a la red. Es decir, entrega un mayor poder a aquellos usuarios que tienen mayor tiempo en la red, demostrando un buen comportamiento.

3.2.3. Ataques a la Estructura de la cadena

Bifurcaciones

Una bifurcación representa una condición en la que, los nodos de la red tienen puntos de vista divergentes sobre ese estado de *Blockchain* que persiste durante largos períodos de tiempo o incluso de forma indefinida. [62]

Las bifurcaciones se pueden crear por fallas de protocolos o incompatibilidad en las actualizaciones de *software* cliente. A lo largo de la vida de Bitcoin su estructura ha sufrido aproximadamente nueve principales bifurcaciones entre el año 2009 y el 2017 y cada bifurcación es tomada como una nueva aplicación o variante de Bitcoin.

Las bifurcaciones también pueden ser causadas con intenciones maliciosas, como la implantación de "nodos Sybil". Representan un estado inconsistente que puede ser explotado por adversarios para causar confusión, transacciones fraudulentas y desconfianza dentro de la red. [62] [63]

- **Implicaciones:** Puede causar división de la cadena y pérdida de ingresos.
- **Contramedidas:** Consenso en conjunto de la mayoría de nodos de la red.

Bloques obsoletos

Los bloques obsoletos ocurren principalmente en las cadenas de bloques públicas. Durante el proceso de validación, dos o más mineros pueden encontrar una solución válida para el

bloque, la red finalmente acepta uno de los bloques ganadores y descarta el resto. Como resultado los bloques no aceptados se convierten en bloques obsoletos ya que no se adjuntan a la *Blockchain* principal. [62]

Bloques huérfanos:

Es un bloque cuyo campo *hash* del bloque padre apunta a un bloque no auténtico que está separado de la *Blockchain*. Los bloques huérfanos se encuentran con mayor frecuencia en las criptomonedas donde el tiempo promedio de cálculo de bloques es pequeño.[62]
Los bloques huérfanos también pueden ocurrir debido a retrasos impredecibles en la propagación del bloque. Es posible que un bloque válido no llegue a la mayoría de los pares de la red debido a cambios en la red y retrasos en la propagación. [62]

Los bloques huérfanos se almacenan de forma temporal en un *pool* denominado *"orphan block pool"*. Las transacciones incluidas no se pierden. En su lugar, otro nuevo bloque las incluye y confirma. Esto es algo que puede suceder en el bloque de la cadena más larga o unos minutos después en el bloque siguiente. [64]

Por lo tanto, el comportamiento de la red y la distribución del retardo también pueden afectar el número de bloques huérfanos en un sistema *Blockchain*.

Para los bloques obsoletos y bloques huérfanos se consideran: [63]

- **Implicaciones:** Puede causar pérdida de ingresos.
- **Contramedidas:** Incrementar el tiempo entre bloques

3.2.4. Ataques al sistema Peer-to-Peer (P2P)

El utilizar un sistema descentralizado P2P es lo que proporciona garantías a la tecnología *Blockchain,* pero también puede contribuir a hacerla vulnerable a ciertos ataques, los cuales exploraremos a continuación.

Selfish Mining

El ataque de minería egoísta o *selfish mining attack* (por sus siglas en ingles), ocurre cuando un minero mantiene sus bloques en privado para obtener una cadena más larga que la *Blockchain* pública. Una vez que la *Blockchain* pública comienza a acercarse a la longitud de su cadena privada, los mineros egoístas liberan sus bloques para reclamar recompensas en bloque. [62]

Esta práctica es deshonesta ya que saca ventaja del sistema y no permite que los mineros honestos obtengan sus recompensas, aun cuando hayan completado el bloque inician antes que el minero egoísta.

Aunque el esfuerzo de cálculo realizado por el minero honesto se desperdicia, su bloque no es eliminado de la red *Blockchain*, lo que conduce a otro problema importante, en la red que es el bloque obsoleto visto anteriormente. [62]

El incentivo para adoptar esta estrategia de minería egoísta es maximizar las recompensas en bloque mediante la publicación de una cadena más larga.

- **Implicaciones:** Puede causar pérdida de ingresos y minería mal intencionada.[63]
- **Contramedidas:** Utilizar *time-stamping-blocks*, para determinar el momento exacto en el que el bloque ha sido minado y validado por la red *Blockchain*. [63]

Ataques de Red

Los ataques asociados a la red *Blockchain* incluyen, entre otros, los ataques DNS (del inglés *Domain Name System*), ataques *BGP* (del inglés *Border Gateway Protocol*) y los ataques Eclipse. Para cada uno de estos ataques, el objetivo del atacante es aislar a los usuarios y mineros de la red real, limitar su acceso a los recursos de la red o crear una partición en la red.

- **Ataques de *DNS* (del inglés *Domain Name System*):** El atacante envenena la caché de DNS y modifica los datos. Cuando un usuario consulta al servidor para obtener direcciones IP de los pares que aceptan conexiones, se lo enruta a la red del atacante. El atacante puede engañar al usuario proporcionándole bloques y transacciones falsas. [62]

- **Ataques BGP (del inglés *Border Gateway Protocol) hijacks*:** Consiste en la modificación de forma no autorizada y con fines maliciosos de las tablas de enrutamiento del protocolo BGP. Puede causar pérdida de ingresos, particionado de la cadena y robos.[63]
- **Ataques de Eclipse:** Consiste en que un grupo de nodos maliciosos aísla sus nodos vecinos utilizando direcciones IP, comprometiendo así su tráfico entrante y saliente. Con suficientes nodos comprometidos en un clúster, el atacante puede aislar nodos válidos y cambiar su vista de *Blockchain*, puede controlar su tráfico entrante y saliente y alimentarlos con información falsa sobre *Blockchain* y transacciones. [62]

Ataques de *DDoS* (por sus siglas en inglés, *Distributed Denial of Service*)

La tecnología *Blockchain*, y a pesar de ser un sistema de igual a igual, sigue siendo propensa a los ataques *DDoS*. Las aplicaciones basadas en *Blockchain*, como Bitcoin y Ethereum, han sufrido repetidamente estos ataques.[62]

El atacante utiliza diferentes dispositivos los cuales realizan peticiones de manera simultánea hacia una misma dirección destino, con el fin de inhabilitar a la víctima. Una contramedida es incrementar el tamaño del bloque.[63]

***Block Withholding Attacks* o Ataques de Retención del Bloque**

En el ataque de retención de bloques, un minero malicioso en el *pool* de minería, resuelve el *PoW* y elige no revelarla al operador del pool. El resto de los mineros en el grupo desperdician sus recursos para encontrar el *nonce* y eventualmente pierden la carrera. [62]

El minero malintencionado puede confabularse con otros grupos de minería y compartir el *PoW* con ellos para obtener una recompensa más alta, o incluso publicar el bloque de forma independiente con una identidad diferente. Debido a este comportamiento injusto, todo el grupo se ve privado de las recompensas en el bloque.[62]

Consensus Delay

Otro ataque asociado con la arquitectura de la naturaleza *peer-to-peer* es el retraso del consenso. En este un atacante puede inyectar bloques falsos u obsoletos para agregar

latencia a la red y evitar que los pares lleguen a un consenso sobre el estado de *Blockchain*. [62]

Los retrasos en la transmisión están sujetos al tamaño del bloque y los mensajes, mientras que los retrasos en la propagación dependen del ancho de banda del enlace entre los nodos.

En tales condiciones, se pueden introducir retrasos intencionales en la red mediante la propagación de bloques obsoletos o transacciones de doble gasto. Los nodos que no son conscientes de los bloques obsoletos responderán con mensajes de *getdata* y al recibir el bloqueo, perderán tiempo en su verificación.[62]

Este tipo de ataques puede causar retrasos y perdida de información, una contramedida puede ser implementar monitorización de los nodos. [63]

Ataques de *Timejacking*

En una cadena *Blockchain* distribuida, todos los nodos deben estar sincronizados en fecha y hora, es por ello que cada nodo posee un contador interno. En este caso un atacante podría adelantar o atrasar dicho reloj de "sistema" realizando diferentes conexiones suplantando a varios nodos y enviando una estampa de tiempo diferente e incorrecta en cada conexión.[63]

Afecta a los mineros grupos de minería y aplicaciones, puede causar división de la cadena, pérdida de ingresos, minería malintencionada y retrasos.[63]

3.2.5. Ataques a la Aplicación de *Blockchain*

Las aplicaciones *Blockchain* tienen sus propias vulnerabilidades y superficie de ataque, principalmente en aplicaciones como las criptomonedas y los contratos inteligentes. Expondremos a continuación las que consideramos importante de tratar en una red C2C

Blockchain Ingestion y Anonimato

Las *Blockchain* públicas brindan accesibilidad de datos abiertos al público, por lo que pueden revelar información útil a un atacante que utilice herramientas de análisis. Este proceso se conoce como ingestión de *Blockchain*. [62]

El anonimato en el uso de criptomonedas basadas en *Blockchain* brinda oportunidades lucrativas para que los delincuentes lleven a cabo actividades fraudulentas. Como tal, las criptomonedas se han convertido en una fuente popular de transferencia de fondos para actividades ilícitas asociadas con la Deep Web. [62]

La ausencia de una autoridad central hace que sea más difícil reclamar un fraude y esperar un reembolso. Por lo tanto, las aplicaciones *Blockchain* pueden ser explotadas para facilitar los delitos cibernéticos y los fraudes en línea. [62]

Doble Gasto

Esta vulnerabilidad se describió en el apartado 3.2.2. En el doble gasto, hay dos transacciones derivadas de la misma salida de transacción no gastada del remitente, y solo una de ellas se incorpora a *Blockchain*. Afecta a la red y a los usuarios. [62]

Cryptojacking

Cryptojacking es una forma de ataque que se lanza en servicios web y basados en la nube para realizar *PoW* ilegalmente. [62]

Implica secuestrar un dispositivo objetivo para realizar cálculos de *PoW* para el atacante. Inicialmente, estos ataques se lanzaron contra proveedores de servicios en la nube, donde usuarios malintencionados realizaban operaciones de minería encubiertas en máquinas virtuales y agotaron los recursos de la nube.

En el *criptojacking* basado en navegador, el navegador web del dispositivo cliente ejecuta código JavaScript que establece una conexión *WebSocket* con un servidor *dropzone* remoto. Luego, el servidor envía el objetivo al cliente, que calcula los valores *hash* para *PoW* y los transmite de vuelta al servidor. Durante este proceso, el propietario del dispositivo sigue sin conocer esta actividad en segundo plano. [62]

El *cryptojacking* en el navegador no solo representa una gran amenaza para la privacidad, sino que también perjudica el rendimiento del dispositivo visitante, ya que los cálculos de *hash*

basados en *PoW,* requieren un uso intensivo del procesador y pueden provocar un uso excesivo de la CPU y el consumo de energía. [62]

Robo de billetera

Un problema bien conocido en las criptomonedas basadas en *Blockchain* son la exposición y el robo de claves privadas. Si el atacante adquiere la clave privada que pertenece a un usuario, puede firmar y generar una nueva transacción en nombre del usuario, y posiblemente gastar su saldo en destinatarios no autorizados.

Este tipo de ataque incluye a todas las billeteras digitales o físicas, y el robo se puede materializar de múltiples formas.[63]

Ataques en contratos inteligentes

La explotación de errores de la aplicación basada en *Blockchain*, surge cuando hay un error en el código del contrato inteligente (*Smart Contracts*). Surge cuando los desarrolladores no identifican errores de código en la aplicación descentralizada. Los atacantes pueden drenar todo el dinero de la billetera del contrato a través de simples errores de código.

La aplicación de contrato inteligente más conocida en el mundo digital es Ethereum. Algunos de los ataques más conocidos contra los contratos inteligentes de Ethereum incluyen:

Ataques de reentrada: Se produce cuando el usuario no actualiza el saldo antes de enviar Ether, eN ese caso un atacante podría robar todo el Ether, haciendo llamadas recursivas al método *call.value()* en un token. [63]

Ataques *DoS* (por sus siglas en inglés, *Denial of Service*): El ataque *DoS* en un *Smart contracts* vulnerable, puede causar pérdida de ingresos, retrasos y robos. Por ejemplo, en una subasta un postor malintencionado intenta convertirse en el líder y cancela todas las licitaciones enviadas por otros postores y mantiene al postor como el líder de la subasta durante el tiempo que quiera. [62]

Ataques de desbordamiento

Ocurre con los contratos inteligentes cuando el valor máximo de un variable es sobrepasado (2^{256}). Por ejemplo, en un contacto inteligente de apuestas online, si alguien envía gran cantidad de ether, superando (2^{256}), el valor de la apuesta se establecería en 0. [62]

Aunque el intercambio de un valor de ether mayor que (2^{256}) no es realista, pero sigue siendo una vulnerabilidad de programación en los contratos inteligentes escritos en código *Solidity*. [62]

Ataques de direcciones cortas: Las vulnerabilidades del ataque de dirección corta un error en la máquina virtual de Ethereum para hacer tokens adicionales en compras limitadas. [62]

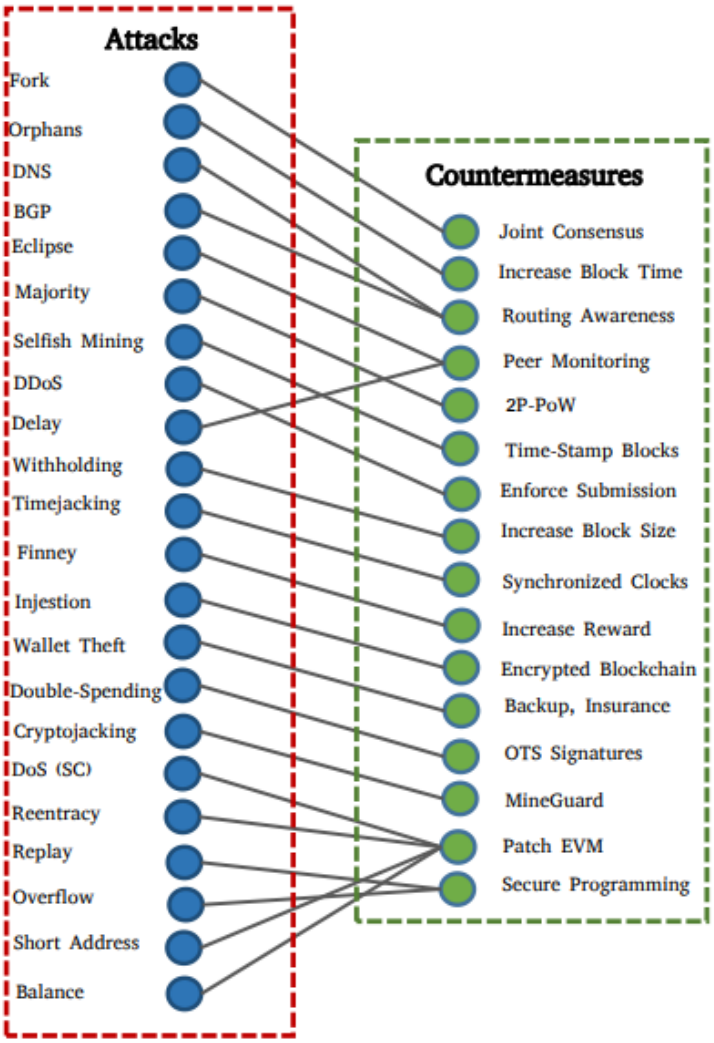
Se aplica principalmente a los tokens, el atacante crea una billetera Ethereum que termina en el dígito "0". Luego realiza una compra en la dirección quitando el último "0". Si el contrato tiene un saldo suficiente, entonces la función de compra no verifica la dirección del remitente y por cada 1000 tokens comprados, la máquina devuelve 256000 tokens. [62]

Transferencia de saldo forzoso: cuando una criptomoneda se bifurca en dos monedas separadas. Un usuario tiene la opción de realizar una transacción en cualquiera de las dos cadenas. [62]

En los ataques de repetición, el atacante olfatea los datos de la transacción en un libro mayor y los reproduce en el otro libro mayor. Como tal, el usuario pierde activos en ambas cadenas. Por lo tanto, una transacción realizada en una red de prueba se puede replicar en la red pública para robar fondos.

Existen diferentes contramedidas para prevenir los ataques o vulnerabilidades anteriormente expuestas, se ha tomado como referencia el artículo *"Exploring the Attack Surface of Blockchain: A Systematic Overview*" [62] donde muestra la relación entre varios ataques a *Blockchain* junto con sus contramedidas. Algunos ataques tienen contramedidas comunes que proporcionan direcciones futuras hacia una solución común.

Imagen 25: Ejemplo ataque de desbordamiento



***Fuente*: https://arxiv.org/pdf/1904.03487.pdf**

4. Propuesta de Modelo de Comercio Electrónico C2C con base en Tecnología *Blockchain*

4.1. Descripción del modelo

En un primer acercamiento, el modelo propuesto de comercio electrónico sobre red *Blockchain* pública *C2C*, toma como alcance el registro de la información de usuarios y de productos, conformando un ecosistema donde compradores y vendedores pueden interactuar sin intermediarios.

Con base al estado del arte recolectado y la identificación de las debilidades de una red *C2C* tradicional, el modelo propuesto a continuación se basa en el uso de una red *Blockchain* de tipo pública *P2P*, que tenga la capacidad de gestión y ejecución de contratos inteligentes.

De acuerdo a las características y funcionalidades de las principales *Blockchain* del mercado, se determinó que *Blockchain* Ethereum es la mejor candidata, para implementar dicho modelo. Dado que incorpora la capacidad de desarrollo de contratos inteligentes personalizados y se implementa como red pública, idóneo para los contextos C2C.

El modelo consta de 7 procesos para el registro de usuario y 5 procesos para el registro de productos.

En cada proceso se hace uso de las siguientes herramientas y/o mecanismo que fortalecen la integridad y autenticidad de la red Blockchain

- Aplicación de registro C2C
- Usuarios y productos.
- Nodos de autoridad
- Masternodes (Usuarios Revisores)
- Smart Contract
- Cifrado de claves
- Funciones hash
- Estampa de tiempo
- Métodos de Consenso

Proceso para el Registro de Usuarios

a) El usuario realizaría una primera interacción, habilitando la Aplicación de registro C2C en su dispositivo. Es posible que se requiera realizar un pago para adquirir la aplicación.

b) Habilitación de uno o más nodos, una vez descargada la aplicación el dispositivo del usuario se conecta a la red *Blockchain*. Este nodo local se definirá como la puerta de enlace del nuevo usuario con toda la red, y contendrá una copia de la red al momento sincronizarse con la red *Blockchain*.

c) Se inicia el proceso de registro de usuario, utilizando la infraestructura de nodo local, el usuario podrá disponer de un formulario de registro el cual le solicitará la información que se registrará en la *Blockchain*.

   Esta funcionalidad del nodo local permite una integración directa entre el usuario y la red *Blockchain*. La información mínima solicitada para el alta de usuarios comprende la relacionada con la identificación de la persona y que posteriormente será complementada con validaciones biométricas. Al finalizar este paso se entregará un par de llaves públicas - privadas, las cuales se utilizará para participar en la red y que serán gestionadas por la aplicación.

   Con relación a los procesos de biometría, lo definimos como proceso complementario al registro del usuario desde el nodo local. Se ha integrado este elemento como un punto de control adicional para determinar que la persona a registrar es realmente quien dice ser. Este proceso de biometría puede comprender validaciones de fotografías tipo *selfie* y prueba de vida.

d) La red *Blockchain* se encargará de la administración de información de los usuarios, pero es importante mantener la privacidad e integridad de la información a lo largo del tiempo, para ello se hará uso del método *proof of existence*

   Proof Of Existence: Consiste en tomar el hash de algo y almacenar ese *hash* en *Blockchain, se puede utilizar para verificar la existencia de un archivo en particular en un momento específico sin compartir el archivo o su contenido en sí.*

   Al momento de registrar un nuevo nodo la red creara un archivo con el código de identificación del nuevo nodo, así como una estampa de tiempo con la fecha y hora en

que el usuario se unió a la red, a este archivo se le aplica el método *proof of existence* para garantizar su autenticidad, posteriormente el resultado de este método se utilizara para validar las transacciones del usuario.

e) La prueba de existencia no implica revelar el contenido del archivo, pues solo se incluye un resumen del mismo. Cualquier variación en el archivo, por pequeño que sea, generaría un resumen criptográfico distinto. Con esto se evitará usurpación y duplicidad de usuarios.

f) *Smart Contract* Registro de Usuarios

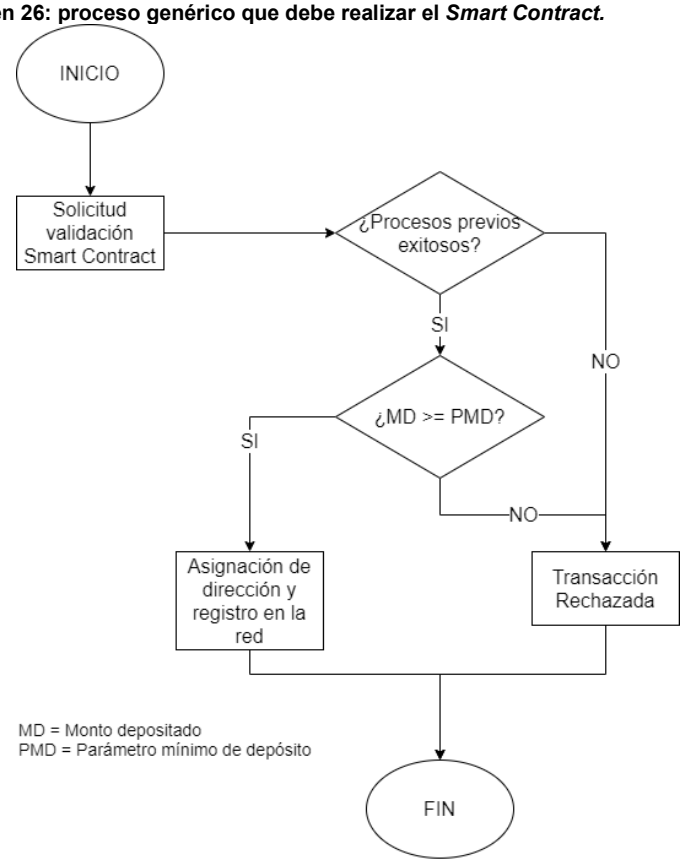
Si la validación de las pruebas biométricas es exitosa, podrá continuar con el proceso de registro de usuario en la *Blockchain*.

Antes de que el usuario pueda iniciar con transacciones de compra y venta, es necesario implementar un mecanismo anti fraude, para ello se configuraran *smart contract* en la red Blockchain, esto consta de pedir al usuario una cuota de participación, también conocido como *proof of stake*.

La red solicitará al usuario un monto inicial el cual será retenido por la red, si el usuario incumple las reglas de compra-venta estipuladas, este será penalizado perdiendo por completo la cuota de partición y el usuario será removido de la red. El objetivo es garantizar que los tratos entre usuarios sean justos y evitar irregularidades o actividades ilícitas dentro de la red.

El siguiente diagrama ejemplifica el proceso genérico que debe realizar el *Smart Contract.*

Imagen 26: proceso genérico que debe realizar el *Smart Contract.*



***Fuente*: Autoría propia**

g) Durante este paso entran en acción los *Masternodes* de la red, los cuales hacen un *checklist* de validación de la información proporcionada por el usuario, si la información es correcta y autenticada, el registro es exitoso y el nuevo usuario es añadido a la red, de lo contrario, es rechazado.

h) Si el registro del nuevo usuario es exitoso, los nodos de autoridad se encargarán de publicarlo y propagarlo a todos los nodos restantes dentro de la red Blockhain, hasta que todos los nodos estén sincronizados.

El proceso anteriormente descrito se repetirá para cada nuevo usuario.

Proceso para el Registro de Productos

Cualquier usuario de la red *Blockchain C2C* registrado exitosamente, podrá hacer uso del rol de comprador o vendedor en cualquier momento.

a) El primer paso es ser un usuario autenticado y registrado en la red *Blockchain*. El cual debe contar con su juego de llaves público y privadas.

b) El usuario ingresa a la plataforma C2C cada uno de sus productos, para ello debe ingresar una fotografía del producto, descripción o ficha del producto que contenga las características y funcionalidad del mismo, así como detalle de garantías y precio de venta.

c) Al momento de registrar un nuevo producto, la aplicación C2C creara un archivo con el código de identificación del producto, la descripción y precio definido por el vendedor. A este archivo se le aplica el método *proof of existence* para garantizar su autenticidad, posteriormente el resultado de este método se utilizará para validar las transacciones del vendedor.

d) Una vez que el vendedor ingresa sus productos entra el proceso de verificación de precios, se hace uso de *smart contract* que garantizan que el precio ofertado por el vendedor sea congruente con el producto y su descripción, marca y funcionalidades. Esto se lograría con métodos de estadística y mecanismos de consenso.

   Ya que no existe una entidad que regule los precios ofertados, la misma red debe garantizar que las reglas de oferta y demanda se cumplan y garantizar transacciones justas para todos los miembros de la red.

e) Dentro de la aplicación C2C se busca crear una comunidad de confianza, los usuarios podrán evaluar a un vendedor y sus productos posteriores a una transacción de compra.

   El objetivo es fomentar el respeto y confianza entre todos los que conforman la red, el comprador también podrá dejar comentarios breves que serán públicos, de esta manera los futuros compradores tendrán una referencia del vendedor.

   Durante este paso entran en acción los *Masternodes* de la red, los cuales hacen un *checklist* de la información proporcionada por el vendedor y el precio del producto, si la información es correcta y autenticada, el registro es exitoso y el nuevo producto es añadido a la red, de lo contrario es rechazado.

f) Si el registro del nuevo producto es exitoso, los nodos de autoridad se encargarán de publicarlo y propagarlo a todos los nodos restantes dentro de la red *Blockhain*, hasta que todos los nodos estén sincronizados.

   El proceso anteriormente descrito se repetirá para cada nuevo producto.

4.2. Ventajas del modelo propuesto sobre modelos tradicionales de *C2C*

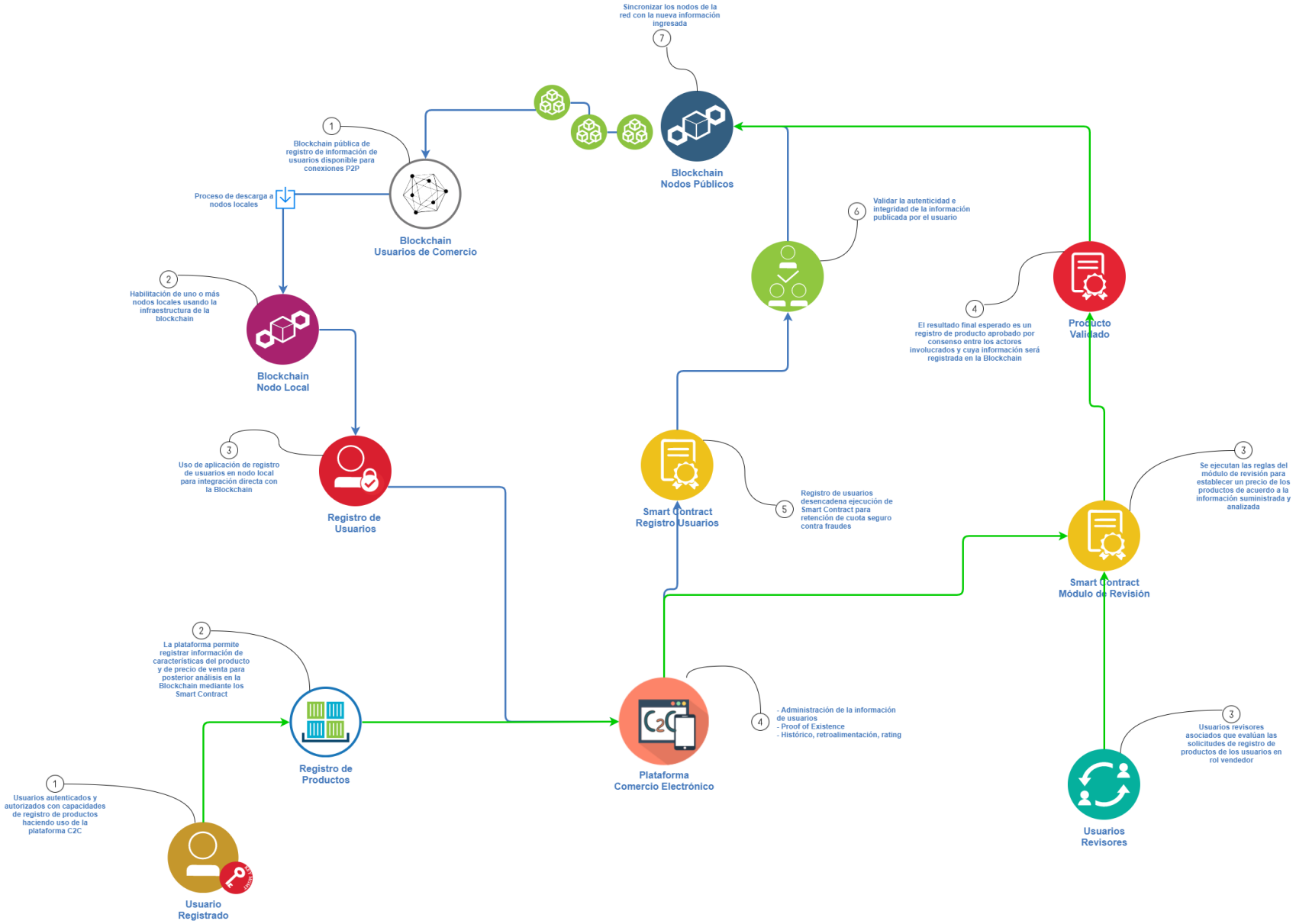
Clasificación	Comercio Electrónico Tradicional	Comercio Electrónico basado en *Blockchain*
Arquitectura de red	Tipo de red controlador/periféricos o cliente/servidor orientado a la centralización	Red de conexión distribuida de tipo *P2P (peer to peer)* para implementar descentralización
Derecho de registro y método de registro	Existe un nodo central que registra y mantiene los datos de forma interactiva	Existen nodos distribuidos que se basan en algoritmos de consenso que determinan el derecho de los registros y el mantenimiento de los datos de forma conjunta
Modo de transacción	Las transacciones necesitan ser confirmadas y mantenidas por un nodo central supervisor	Las transacciones son realizadas sobre red punto a punto que funcionan como testigos y supervisión colectiva
Relación de confianza	El nodo central establece una aprobación de confianza para todos los nodos	La relación de confianza se basa en la auto certificación de nodos mediante el cifrado asimétrico verificando la identidad y prueba de conocimiento cero para verificar la información.
Fraude en las transacciones	Existe la probabilidad de fraude en el nodo central, lo que lo convierte en un riesgo potencial.	La posibilidad de fraude se reduce dado que existe almacenamiento distribuido y mediante la implementación de algoritmos de consenso
Manipulación de la Información	La posibilidad de que el nodo central sea comprometido implica que la data pueda ser manipulada y repudiada	La capacidad de almacenamiento distribuido, estructuras de datos relacionadas, marcas de tiempo y algoritmos hash reducen la posibilidad de manipulación y repudio.

Fiabilidad del almacenamiento de datos	El almacenamiento y respaldos de recuperación ante desastres son basados en el nodo central	El sistema puede continuar operando si algunos nodos fallan. La información de los nodos fallidos puede ser recuperada.
Protección de la privacidad	Los participantes necesitan proveer información de identidad personal que es almacenada en el nodo central. A partir de este punto, la información de identidad puede experimentar fugas, ataques o secuestros.	Los participantes son identificados en la *Blockchain* por su ID cifrado. Por la seguridad de la red, la información de las partes no será divulgada.
Economía en las transacciones	Dado que existe una plataforma centralizada, los costos se incrementan	La *Blockchain* permite una interacción directa a través de procesos simplificados, lo que conlleva en costos bajos,

Tabla 8: Ventajas del modelo propuesto

4.3. Arquitectura del modelo

Imagen 27: Arquitectura del modelo.



***Fuente*: Autoría propia**

Referencias

Número	Referencia
1	R. T. Wigand, Electronic Commerce: Definition, Theory, and Context, Arizona: Arizona State University, 1997.
2	M. Kütz, Introduction to E-Commerce, United States: Bookboon, 2016.
3	V. Zwass, Electronic Commerce and Organizational Innovation: Aspects and Opportunities, Spring: International Journal of Electronic Commerce , 2003.
4	M. Y. &. C. R. T. Kiang, A Framework for Analyzing the Potential Benefits of Internet Marketing. J., Estados Unidos, 2001.
5	A. L. Albertin, O comércio eletrônico evolui e consolida-se no mercado brasileiro, Brasil: Revista de administración, 2000.
6	D. Black, What is electronic commerce, United Kingdom: University o Cardiff, 2000.
7	P. Timmers, "Business Models for Electronic Markets", Semanticscholar.org, 1998. [Online]. Available: https://www.semanticscholar.org/paper/Business-Models-for-Electronic-Markets-Timmers/df8bf13f58c65394fdb1353957c1b90edf45fac4. [Accessed: 19- Mar- 2021].
8	I. Gil and P. Conesa, "MODELOS DE NEGOCIO", Ignaciogil.eu, 2017. [Online]. Available: http://ignaciogil.eu/textos/pes/Modulo3_Modelo_Negocio.pdf. [Accessed: 20- Mar- 2021].
9	A. Al-Alawi and S. Al-Bassam, "The Implications of Unethical and Illegal Behavior in the World of E-Commerce The Implications of Unethical and Illegal Behavior in the World of E-Commerce", ResearchGate, 2019. [Online]. Available: https://www.researchgate.net/publication/335821863. [Accessed: 19- Mar- 2021].
10	C. Rodríguez Merino, "Modelos de negocio y ventajas del E-comerce - Marketing Digital", UPF Barcelona School of Management, 2015. [Online]. Available: https://marketingdigital.bsm.upf.edu/modelos-negocio-ventajas-del-e-comerce/. [Accessed: 18- Mar- 2021].
11	J. Hernandez Hernandez, "El comercio electrónico y sus modelos de negocio en mexico", Repository.usta.edu.co, 2018. [Online]. Available: http://repository.usta.edu.co/bitstream/handle/11634/10899/2018jeimyhernandez.pdf?sequence=1&isAllowed=y. [Accessed: 19- Mar- 2021].
12	S. De Paz Portillo, L. Salgado Quintanilla and J. Tutila Argueta, ""“OPERACIONES DE COMERCIO ELECTRÓNICO Y SU INCIDENCIA TRIBUTARIA”", Ri.ues.edu.sv, 2010. [Online]. Available: http://ri.ues.edu.sv/id/eprint/11492/1/D419o.pdf. [Accessed: 18- Mar- 2021].
13	H. VARGAS OLIVARES, "MODELOS DE E-BUSINESS", Docencia.fca.unam.mx. [Online]. Available: http://docencia.fca.unam.mx/~gcervantes/blog/Modelos_E-Business_Hec.pdf. [Accessed: 20- Mar- 2021].

14 J. Miguela, "C2C Definición y características. Ventajas para los compradores.", Slideplayer.es, 2014. [Online]. Available: https://slideplayer.es/slide/137852/. [Accessed: 20- Mar- 2021].

15 E. Business, "E-Commerce: ¿Cómo funciona el modelo de comercio electrónico C2C?", Esan.edu.pe, 2015. [Online]. Available: https://www.esan.edu.pe/apuntes-empresariales/2015/05/e-commerce-como-funciona-modelo-comercio-c2c/#:~:text=En%20el%20comercio%20electr%C3%B3nico%20C2C,son%20quienes%20toman%20la%20batuta.&text=En%20esta%20transacci%C3%B3n%20no%20intervienen,servicios%20sino%20los%20mismos%20consumidores. [Accessed: 20- Mar- 2021].

16 T. Prezzavento, "Cambridge Analytica y su Impacto en las Políticas de Privacidad." Available: https://repositorio.udesa.edu.ar/jspui/bitstream/10908/16625/1/%5BP%5D%5BW%5D%20T.L.%20Com.%20Prezzavento%2C%20Timoteo.pdf. [Accessed 6 January 2021].

17 "¿Qué es la propiedad intelectual?", Wipo.int, 2021. [Online]. Available: https://www.wipo.int/about-ip/es/. [Accessed: 06- Mar- 2021].

18 "El Salvador, el segundo de América Latina con menos conectividad a internet - Diario El Mundo", Diario El Mundo, 2021. [Online]. Available: https://diario.elmundo.sv/el-salvador-el-segundo-de-america-latina-con-menos-conectividad-a-internet/. [Accessed: 06- Feb- 2021].

19 F. Oliva and F. Dadalt, "En la búsqueda de la Omnicanalidad, el cliente en el centro nuevamente", Deloitte Vision, 2016. Available: https://www2.deloitte.com/content/dam/Deloitte/uy/Documents/technology/Articulo%20Omnicanalidad.pdf. [Accessed 13 March 2021].

20 P. Simon, "Desafíos del Comercio Electrónico y el Camino hacia la Omnicanalidad", Trabajo de fin de máster, Universidad Torcuato Di Tella, 2017.

21 W. Viriyasitavat and D. Hoonsopon, "Blockchain characteristics and consensus in modern business processes", Journal of Industrial Information Integration, vol. 13, pp. 32-39, 2019. Available: 10.1016/j.jii.2018.07.004 [Accessed 25 April 2021].

22 "Blog sobre negocio y marketing digital | ISDI", Isdi.education, 2021. [Online]. Available: https://www.isdi.education/es/isdigital-now/como-revolucionaran-impresoras-3d-e-commerce. [Accessed: 25- Mar- 2021].

23 H. J. Fúquene Ardila y L. Aroca B, «Comercio electrónico y realidad aumentada: una gran alianza», Rev. vínculos, vol. 11, n.º 1, pp. 172–179, dic. 2014.

24 *S. Madakam and S. Kollu, "Blockchain Technologies Fundamentals - Perceptions, Principles, Procedures and Practices", PRAJNAN - Journal of Social and Management Sciences, no. 0970-8448, 2020.*

25 Walker, W., 2018. Blockchain: Aplicaciones y Entendimiento En El Mundo Real. [online] Google Books. Available at: [Accessed 15 April 2021].

26 W. Navas Bayona, H. Loor Zambrano and C. Amen Chinga, "LA CONSOLIDACIÓN DEL BLOCKCHAIN EN LAS EMPRESAS COMO MÉTODO DE PAGO PARA SUS TRANSACCIONES", *Investigación &*

Negocios, vol. 13, no. 22, p. 135, 2020. Available: 10.38147/invneg.v13i22.108 [Accessed 6 February 2021].

27 L. Moreno González and A. Mesa, "Condiciones de uso y aplicación de la tecnología Blockchain para los productos financieros autorizados en Colombia", *Repository.unaula.edu.co*, 2020. [Online]. Available: http://repository.unaula.edu.co:8080/jspui/handle/123456789/1406. [Accessed: 06- Feb- 2021].

28 W. Viriyasitavat y D. Hoonsopon, «Blockchain Characteristics and Consensus in Modern Business Processes,» *Journal of Industrial Information Integration,* p. 13, 2018.

29 S. Nakamoto, «Bitcoin: A Peer-to-Peer Electronic Cash System,» 23 06 2018. [En línea]. Available: https://bitcoin.org/bitcoin.pdf.

30 M. Allende Lopez y V. Colina Unda, «Inter-American Development Bank,» 28 06 2018. [En línea]. Available: https://blogs.iadb.org/conocimiento-abierto/es/elementos-clave-de-blockchain/.

31 B. Academy, «Binance Academy,» Binance Academy, 7 8 2018. [En línea]. Available: https://academy.binance.com/es/articles/what-are-nodes.

32 A. Narayanan, J. Bonneau, E. Felten, A. Miller y S. Goldfeder, *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction,* Princento University Press, 2016.

33 J. Caballero, *Estudio de tecnologías Bitcoin y Blockchain.,* España: Universidad Oberta de Catalunya., 2018.

34 N. I. O. S. a. Technology, *Blockchain Technology Overview,* Gaithersburg (Maryland) , 2018.

35 B. Academy, «Bit2me Academy,» 06 06 2019. [En línea]. Available: https://academy.bit2me.com/en/what-is-a-pruned-node/. [Último acceso: 20 06 2021].

36 T. J. Rush, «Thomas Jay Rush,» 13 08 2020. [En línea]. Available: https://tjayrush.medium.com/building-your-own-ethereum-archive-node-72c014affc09. [Último acceso: 20 06 2021].

37 B. Magazine, «Nasdaq,» Bitcoin Magazine, 01 03 2021. [En línea]. Available: https://www.nasdaq.com/articles/how-to-operate-a-profitable-lightning-node-2021-03-01. [Último acceso: 20 06 2021].

38 *Ley para Regular las Insituciones de Tecnología Financiera,* Mexico, 2018.

39 B. d. México, *Activos Virtuales,* México, 2018.

40 C. D. Retamal, *La Blockchain: Fundamentos, aplicaciones y relación con otras tecnologías disruptivas,* España: Universitat Politécnica de Catalunya, 2010, p. 1/8.

41 b. ACADEMY, "Transacciones Bitcoin ¿Cómo funcionan y qué debes tener en cuenta?", Bit2Me Academy, 2021. [Online]. Available: https://academy.bit2me.com/transacciones-bitcoin/. [Accessed: 15- May- 2021].

42 "Bloques y transacciones", Blockchain Federal Argentina. [Online]. Available: https://bfa.ar/blockchain/bloques-y-transacciones. [Accessed: 15- May- 2021].

43 X. Xu, I. Weber and M. Staples, Architecture for block chain applications, 1st ed. Switzerland: Springer Nature, 2019, pp. 15-16.

44 N. Rodriguez, "Algoritmos de consenso: la raíz de la tecnología blockchain", 101Blockchains, 2018. [Online]. Available: https://101blockchains.com/es/algoritmos-de-consenso-blockchain/. [Accessed: 25- May- 2021].

45 "Problema de los generales bizantinos - Wikipedia, la enciclopedia libre", Es.wikipedia.org, 2020. [Online]. Available: https://es.wikipedia.org/wiki/Problema_de_los_generales_bizantinos. [Accessed: 25- May- 2021].

46 S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System", Bitcoin.org, 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf. [Accessed: 26- May- 2021].

47 p. university, "Peercoin University", University.peercoin.net. [Online]. Available: https://university.peercoin.net/#/9-peercoin-proof-of-stake-consensus. [Accessed: 26- May- 2021].

48 "Algoritmo de consenso-PBFT (Sistema práctico de tolerancia a fallas bizantinas) - programador clic", Programmerclick.com. [Online]. Available: https://www.programmerclick.com/article/6581269295/. [Accessed: 26- May- 2021].

49 A. Singh, R. Parizi, Q. Zhang, K. Choo and A. Dehghantanha, "Blockchain smart contracts formalization: Approaches and challenges to address vulnerabilities", Computers & Security, vol. 88, p. 101654, 2020. Available: 10.1016/j.cose.2019.101654 [Accessed 16 May 2021].

50 M. ECHEBARRÍA SÁENZ, "CONTRATOS ELECTRONICOS AUTOEJECUTABLES (SMART CONTRACT) Y PAGOS CON TECNOLOGÍA BLOCKCHAIN", *Revista de Estudios Europeos*, no. 2530-9854, pp. 69-97, 2017.

51 "Best Programming Languages to Build Smart Contracts", Blockchain-council.org, 2021. [Online]. Available: https://www.blockchain-council.org/blockchain/best-programming-languages-to-build-smart-contracts/. [Accessed: 18- May- 2021].

52 R. Parizi, A. Singh and A. Dehghantanha, "Smart Contract Programming Languages on Blockchains: An Empirical Evaluation of Usability and Security", 2018. [Accessed 30 May 2021].

53 R. Luque Lodeiro, "Blockchain: Estado del arte, tendencias y retos", Trabajo de Fin de Máster, Universidad de Oviedo, 2020.

54 M. Suárez Taboada, "Desarrollo de una aplicación descentralizada con blockchain: DApp para el acceso y modificación de información sensible", Hdl.handle.net, 2021. [Online]. Available: http://hdl.handle.net/10609/106746. [Accessed: 16- May- 2021].

55 J. Itzep Salvador, "ESTUDIO DE BLOCKCHAIN DESDE LA PERSPECTIVA DE ESTRUCTURAS DE DATOS", INGENIERO EN CIENCIAS Y SISTEMAS, UNIVERSIDAD DE SAN CARLOS DE GUATEMALA, 2019.

56 B. Singhal, G. Dhameja and P. Panda, "How Blockchain Works", *Beginning Blockchain*, pp. 31-148, 2018. Available: 10.1007/978-1-4842-3444-0_2 [Accessed 30 Mayo 2021].

57	A. Shanti Bruyn, "Blockchain an Introduction", University Amsterdan, pp. 19-40, 2017. [Accessed 27 March 2021].
58	"¿Cómo funciona la Cadena de Bloques (Blockchain) ?| Bit2Me Academy", Bit2Me Academy, 2021. [Online]. Available: https://academy.bit2me.com/como-funciona-blockchain-cadena-de-bloques/. [Accessed: 30- Apr- 2021].
59	J. Maldonado, "¿Qué es el nonce? Un número vital en Bitcoin", Cointelegraph, 2021. [Online]. Available: https://es.cointelegraph.com/explained/what-is-the-nonce-a-vital-number-in-bitcoin. [Accessed: 30- Mar- 2021].
60	G. Iredale, "Blockchain vs Database: Understanding The Difference", 101Blockchains, 2020. [Online]. Available: https://101blockchains.com/blockchain-vs-database-the-difference/. [Accessed: 05- Jun- 2021].
61	X. Xu, I. Weber and M. Staples, Architecture for block chain applications, 1st ed. Switzerland: Springer Nature, 2019, pp. 19-20.
62	M. Saad et al., "Exploring the Attack Surface of Blockchain: A Systematic Overview", Arxiv.org, 2019. [Online]. Available: https://arxiv.org/pdf/1904.03487.pdf. [Accessed: 30- Jun- 2021].
63	R. Luque Lodeiro, "BLOCKCHAIN:ESTADO DEL ARTE, TENDENCIAS Y RETOS", Digibuo.uniovi.es, 2020. [Online]. Available: https://digibuo.uniovi.es/dspace/bitstream/handle/10651/56337/TFM_RubenLuqueLodeiro.pdf?sequence=6&isAllowed=y. [Accessed: 30- Jun- 2021]
64	B. Academy, "¿Qué es un bloque huérfano? | Bit2Me Academy", Bit2Me Academy. [Online]. Available: https://academy.bit2me.com/que-es-un-bloque-huerfano/#:~:text=En%20blockchain%2C%20un%20bloque,otras%20criptomonedas%20derivadas%20del%20mismo. [Accessed: 30- Jun- 2021]
65	X. Li, P. Jiang, T. Chen, X. Luo and Q. Wen, "A survey on the security of blockchain systems", Future Generation Computer Systems, vol. 107, pp. 7-10, 2020. Available: https://www.sciencedirect.com/science/article/pii/S0167739X17318332. [Accessed 20 July 2021].
66	P. Druschel, F. Kaashoek and A. Rowstron, Peer-to-peer systems: first International Workshop,IPTPS 2002. Berlin: Springer, 2002, pp. 1-5.
67	¿Qué es un Ataque Sybil? | Bit2Me Academy", Bit2Me Academy. [Online]. Available: https://academy.bit2me.com/que-es-un-ataque-sybil/. [Accessed: 23- Jul- 2021].
68	D. Geroni, "Los 5 principales problemas de seguridad de la cadena de bloques en 2021", *101 cadenas de bloques* , 2021. [En línea]. Disponible: https://101blockchains.com/blockchain-security-issues/. [Consultado: 28 de julio de 2021].