Método de Optimización en Procesos Contractuales Interorganizacionales aplicando Smart Contracts en Plataformas IoT

Mg. Ing. Silvia Poncio

Paulo José Ordóñez Giovanazzi

Trabajo Final de Carrera presentado para obtener el título de

Lic. en Gestión de Tecnología Informática

Noviembre, 2023

Resumen

Dada la creciente adopción de las tecnologías IoT y blockchain en las industrias, en particular en el transporte de activos, se observa que es propicio aplicar Smart Contracts en ciertos escenarios y obtener beneficios considerables entre las organizaciones que se integren a la red. El propósito de este enfoque radica en la optimización de los procesos contractuales entre organizaciones, que a menudo están plagados de trámites burocráticos. Esta optimización se logra capitalizando la infraestructura ofrecida por los dispositivos y plataformas IoT disponibles. En el marco de esta investigación, se introduce un método innovador que busca automatizar los procesos interorganizacionales necesarios para la ejecución de contratos, aprovechando las capacidades de las plataformas IoT. Esto se logra a través de Smart Contracts que operan según agentes definidos en las plataformas IoT de cada organización. Estos contratos automatizan procesos contractuales al establecer condiciones y utilizar datos de la IoT como activadores, asegurando la ejecución precisa de los acuerdos y mejorando la transparencia y seguridad en las transacciones. Al integrar Smart Contracts en las plataformas IoT en entornos organizacionales, se hace posible la trazabilidad de operaciones, fomentando la transparencia y confiabilidad en las transacciones. La eliminación de intermediarios agiliza procesos, reduciendo la complejidad y acelerando operaciones. Esta automatización disminuye la burocracia, lo que se traduce en una ejecución más eficiente de los acuerdos. Además, garantiza un cumplimiento preciso de los términos contractuales, reforzando la seguridad y confianza en todas las partes involucradas.

Palabras clave:

blockchain, internet of things, internet of vehicle, smart contracts

Dedicatoria

Quiero dedicar este trabajo a mi querida madre, mi principal fuente de apoyo a lo largo de mi trayectoria académica. Su aliento constante ha sido mi impulso para alcanzar cada logro.

A mi hermano, con quien comparto la pasión por la tecnología, con debates, reflexiones y un sinfín de momentos valiosos. Juntos hemos crecido, convirtiendo cada discusión en una experiencia enriquecedora.

A mi papá, agradezco sus lecciones valiosas, especialmente en el arte de la comunicación. Sus enseñanzas han sido una guía esencial para mi desarrollo.

Silvia V. Ponce, Directora de la carrera, mi sincero agradecimiento por su dedicación y rápida solución a mis inquietudes. Su orientación ha sido clave para superar obstáculos y alcanzar mis metas.

Finalmente, dedico este trabajo a todos aquellos interesados en esta investigación, enfocada en el apasionante mundo de la blockchain y cómo las organizaciones pueden capitalizar su infraestructura IoT uniendo estas dos tecnologías. Su interés y apoyo me inspiran a seguir contribuyendo y aprendiendo en estos emocionantes campos.

Reconocimientos

Quiero expresar mi sincero agradecimiento a Silvia V. Poncio por su orientación y apoyo continuo durante mi trayectoria académica. Agradezco también a mi director de proyecto de investigación Alejandro M. Hernandez y a los participantes Iván Ordóñez Giovanazzi, Ulises G. Pignatelli por contribuir en el artículo presentado en el CoNaIISI 2023, Carlos Neil y Pablo Andres Audoglio por su valiosa contribución y colaboración en este trabajo.

Un agradecimiento especial a mis padres y a mi hermano Iván por ser mi fuente constante de apoyo y motivación.

Índice General

Índice de Gráficos

Índice de Tablas

Capítulo 1 Introducción

1.1 Introducción

En el contexto del avance de las Smart Cities, se ha logrado solucionar el problema del ancho de banda gracias a la implementación de la tecnología 5G. Esta tecnología proporciona una tasa de transferencia de datos significativamente superior a las versiones anteriores, lo que permite manipular el creciente flujo de información generado por la gran cantidad de dispositivos conectados en una ciudad inteligente.

A medida que el número de dispositivos interconectados (conocidos como "things") sigue aumentando, la trazabilidad incrementa su complejidad. La trazabilidad se refiere a la capacidad de rastrear y monitorear el movimiento de los objetos y las personas en una ciudad inteligente. El constante aumento de dispositivos conectados, generan enormes volúmenes de datos que deben ser gestionados y analizados de manera eficiente para obtener información útil y tomar decisiones con fundamento.

La complejidad radica en la necesidad de implementar sistemas de gestión de datos complejos y algoritmos de análisis avanzados que permitan procesar y analizar estos datos en tiempo real. Independientemente de esto, se debe garantizar la seguridad, la integridad y la privacidad de estos datos, lo que se convierte en un desafío importante.

La industria del transporte ha ido evolucionando a medida que el desarrollo tecnológico iba avanzando, al día hoy podemos observar que existen una variedad de vehículos con algún tipo de dispositivo inteligente integrado en sus sistema electrónico, tales como los vehículos agrícolas, los vehículos de transporte de carga pesada, los automóviles como servicios de traslados personal, etc. Algunos de estos dispositivos inteligentes que se puede destacar son los GPS, sensores de detección, dispositivos actuadores, controladores, en definitiva se puede decir que son things que a través de dispositivos permiten capturar

información del entorno físico y estos datos viajan a un servidor en alguna parte del mundo con el fín de ser manipulados en beneficio de alguna entidad. Esto se conoce como Internet of Thing o Internet de las Cosas (IoT).

Las empresas disponen un monitoreo y control exhaustivo de los sistemas autónomos y es crucial para el buen desempeño del servicio. La mala administración de los things tiene una alta probabilidad de que los costos se incrementen, la posibilidad de que los things sean interceptados y sean vulnerados para sacar algún tipo de provecho, es posible que el thing pierda el control y provoque posibles accidente poniendo en riesgo la integridad de las personas quienes operan sobre los vehículos como también los ciudadanos.

Otra de las características de los vehículos IoT es informar a otros vehículos los posibles tráficos congestionados en tiempo real, lo cual permite a los conductores identificar las posibles rutas óptimas para llegar a destino en tiempo y forma. Detectar cuando el vehículo infligió una multa, como exceso de velocidad en un determinado horario y día de la semana, determinar cuando hubo un desenganche, o apertura de portón y puerta, si el vehículo está siendo remolcado, toda esta información que brinda cada dispositivo IoT la verificación, control y manipulación de los datos por parte de los things deben ser seguros, consistentes y trazables, con el fin de que las operaciones entre diferentes entidades se cumplan a través de un contratos inteligentes, que es lo que se quiere aportar desde este estudio.

1.2 Problemas y Soluciones

Comenzando con las problemáticas, en los sistemas IoT, la mayoría de las comunicaciones entre dispositivos se realiza a través de una arquitectura cliente/servidor o dicho de otra forma una arquitectura centralizada, en consecuencia la autenticación, identificación y otros procesos de seguridad pasan por una autoridad central. La causa

principal de los problemas centralizados en donde los recursos pasan por esta unidad central, es la posibilidad de un único punto de fallo. En segundo lugar todos los dispositivos conectados deben comunicarse a través de internet independientemente de la distancia entre ellos, lo que produce una sobrecarga en el procesamiento de comandos. Además, el enfoque actual para la seguridad de IoT implica altos costos de mantenimiento en términos de servidores en la nube centralizados y otros equipos de red. En definitiva, para migrar la arquitectura centralizada actual de IoT a un enfoque descentralizado se requieren algunas capacidades fundamentales, como la comunicación peer-to-peer, el intercambio de archivos distribuido, la comunicación autónoma de dispositivos, y la eficiencia y la seguridad. (Ali & Ali, 2018, 1-2)

Los desafíos a tener en cuenta son:

El desafío del costo de conectividad

Los altos costos de infraestructura y mantenimiento asociados con grandes granjas de servidores y nubes centralizadas resultan en soluciones de IoT prohibitivamente costosas. Es poco probable que las empresas tengan un buen margen de beneficio debido a varios años de soporte y mantenimiento requeridos incluso para dispositivos IoT baratos. Este costo de soportar y atender miles de millones de dispositivos inteligentes, incluso algo tan simple como mantener servidores y lanzar actualizaciones de software. (Gubbi et al., September 2013, 1645-1660)

El desafío de seguridad y privacidad

La mayoría de las soluciones de IoT en la actualidad son proporcionadas por autoridades centralizadas, ya sea el gobierno, los fabricantes o los proveedores de servicios. Esto permite que estas autoridades obtengan acceso no autorizado para recopilar y analizar los datos de los usuarios. Se están construyendo enfoques de código cerrado (a menudo descritos como seguridad a través de la oscuridad) en el sistema actual. Pero estas soluciones están

obsoletas y se requiere un nuevo enfoque de código abierto (seguridad a través de la transparencia) para llevar el IoT al siguiente nivel. Aunque los sistemas de código abierto pueden ser susceptibles a la explotación y accidentes, es poco probable que los gobiernos u otras instituciones objetivo recopilen datos de usuarios no autorizados. (Dhakal & Cui, 2018, 2)

El desafío de sostenibilidad

Mientras que el usuario tiende a cambiar sus dispositivos informáticos personales como sus teléfonos inteligentes y computadoras cada pocos años, no es el mismo caso para otros dispositivos como automóviles, casas, refrigeradores, cerraduras de puertas, etc. Se espera que estas cosas duren mucho tiempo después de instaladas. En el mundo del IoT, el costo de las actualizaciones de software y el mantenimiento de los productos que ya han sido descontinuados pesará en los balances de las compañías (Dhakal & Cui, 2018, 2).

En 2018, había más de 17 mil millones de dispositivos conectados a Internet, con unos 7 mil millones siendo de IoT. Se espera que para 2025 esta cifra llegue a 35 mil millones. La importancia de esto radica en el crecimiento de dispositivos inteligentes, especialmente con el 5G. Se espera que para 2028, haya más de 5 mil millones de conexiones 5G, con el 80% de ellas utilizando esta tecnología. Esto se debe a dispositivos IoT más asequibles y eficientes que están siendo adoptados por varias industrias. Estos dispositivos generan grandes cantidades de datos que se almacenan para su análisis en la nube. Aunque los modelos actuales de IoT son cerrados, hay interés en compartir datos en una economía compartida, lo que presenta desafíos en el intercambio de datos (“Ericsson Mobility Report,” 2022, 11-15) (Boncea et al., 2019).

Por otro lado, según un informe, la Conectividad Inteligente como la Movilidad Sostenible son los procesos mejorados basados en la colaboración transfronteriza entre las autoridades (United Nations, 2021). El informe plantea los siguientes desafíos:

Estandarización e interoperabilidad: promover el uso de estándares armonizados y especificaciones técnicas que permitan el intercambio de datos sin problemas y la interoperabilidad entre diferentes sistemas y plataformas.

Privacidad y seguridad de datos: abordar las preocupaciones relacionadas con la privacidad y seguridad de los datos en el contexto de la movilidad sostenible y la conectividad inteligente.

El significado de lograr mejorar estos desafíos una es la sostenibilidad que aumenta gracias al intercambio de datos: los conductores disponen de información sobre su comportamiento al volante para mejorar sus hábitos de conducción, pueden evitar los atascos y las ciudades pueden fomentar la movilidad sostenible mediante la combinación de la movilidad multimodal: saber exactamente dónde hay plazas de aparcamiento libres o conocer los horarios de los autobuses ayuda a reducir las emisiones de CO2 de los coches. Esto también ayuda a promover una movilidad optimizada, en la que sectores como la logística tienen un impacto muy positivo (*Connected Car: What Is It and What Is Its Future?*, 2022).

El IoT generalmente se presenta como la tecnología disruptiva para resolver la mayoría de los problemas de la sociedad actual, como ciudades inteligentes, transporte inteligente, monitoreo de la contaminación, atención médica conectada, entre otros (Sisinni et al., 2018, 2).

Hoy en día y en los próximos años la necesidad de conectar un gran número de dispositivos a Internet a bajo costo, con capacidades de hardware limitadas y recursos energéticos (como baterías pequeñas), hacen que la latencia, la eficiencia energética, el costo, la confiabilidad y la seguridad/privacidad sean características más deseables (Akerberg et al., 2011, 410-415).

Existen antecedentes de investigación del uso de blockchains en el ámbito del IoT (Christidis & Devetsikiotis, 2016). Un caso de estudio explora cómo las blockchains

posibilitan una red distribuida entre pares en la que participantes no confiables pueden interactuar sin intermediarios, de forma verificable. Se analizan los Smart Contracts, que residen en la Blockchain y permiten la automatización de procesos multi-etapa. La combinación de Blockchain e IoT facilita el intercambio de servicios y recursos, creando un mercado de servicios entre dispositivos, así como la automatización de flujos de trabajo existentes, lo que reduce tiempos y costos. Los Smart Contracts automatizan procesos complejos. Al combinar estos elementos con dispositivos IoT, se logra una automatización de flujos de trabajo única y criptográficamente verificable, generando ahorros significativos en tiempo y costos.

Ha sido presentada una arquitectura de software que une blockchain y dispositivos IoT para rastrear productos en una Cadena de Suministro (CS) (Ashraf & Heavey, 2023). Utiliza la blockchain Solana para implementar procesos de la CS y almacena datos relacionados en la blockchain. Los dispositivos IoT, como Sigfox y Sensit, capturan información como temperatura y ubicación. La meta es crear una estructura que permita una blockchain genérica para la CS en el futuro. Esta solución aborda problemas de trazabilidad y control en la CS utilizando la combinación de blockchain e IoT. En otros campos, la combinación de estas tecnologías ha demostrado ser exitosa para rastrear productos y garantizar autenticidad. Esto respalda la estrategia presentada en el presente artículo.

La combinación de tecnologías emergentes como blockchain e IoT ha demostrado ser efectiva en otros campos para garantizar trazabilidad y seguridad en procesos y transacciones. Por ejemplo, en la industria alimentaria y farmacéutica se han utilizado estas tecnologías para rastrear la cadena de suministro y prevenir la falsificación (Rauch, 2021). Estas implementaciones exitosas en otros contextos respaldan la estrategia presentada en el presente artículo para abordar problemas en la CS.

A partir de todo el relevamiento de los artículos presentados se evidencia que los sistemas IoT aún necesitan mejorar aspectos de seguridad de los datos, privacidad, costos y proporcionar sostenibilidad en base a las prospectiva tecnológica de esta área.

Con lo cual, se formula la siguiente pregunta del trabajo principal: ¿Cómo se puede desarrollar e implementar un método eficiente y seguro aplicando Smart Contracts en plataformas IoT con el lenguaje de programación Solidity con el fin de mejorar la eficiencia operativa, la seguridad y la transparencia en la gestión de contratos entre organizaciones?

1.3 Propuesta

La importancia de estos desafíos que son propios del IoT se pueden tratar implementando smart contracts y blockchain, ya que estos garantizan la privacidad y seguridad de los datos, como también la estandarización e interoperabilidad.

Se espera que al desarrollar e implementar un método eficiente y seguro aplicando Smart Contracts en plataformas IoT con el lenguaje de programación de Solidity, es posible mejorar significativamente la eficiencia, la seguridad y la transparencia de las operaciones contractuales entre organizaciones.

1.4 Objetivo del TF

1.4.1 Objetivo general:

Desarrollar e implementar un método eficiente y seguro utilizando Smart Contracts en plataformas IoT, por medio del lenguaje de programación Solidity, con el fin de mejorar la eficiencia operativa, la seguridad y la transparencia en la gestión de contratos entre las organizaciones.

1.4.2 Objetivos específicos:

- Investigar y evaluar las ventajas de los Smart Contracts en sistemas IoT aplicados a casos de uso particularmente a vehículos.
- Diseñar y desarrollar para un caso de uso en el lenguaje de programación de Solidity un Smart Contract aplicado a plataformas IoT.
- Evaluar el rendimiento, la eficiencia y la seguridad del sistema implementado.
- Realizar análisis de los resultados obtenidos con respecto a los enfoques tradicionales.
- Proporcionar recomendaciones para la aplicación y la adopción práctica de la solución propuesta en empresas y organizaciones que gestionan flotas de vehículos.

La investigación propone una aplicación práctica de Smart Contracts en el contexto de la integración entre IoT y blockchain, específicamente en el transporte de activos. La aplicación de Smart Contracts en escenarios interorganizacionales busca optimizar procesos contractuales, reducir trámites burocráticos y mejorar la eficiencia en la ejecución de acuerdos. Las organizaciones que adopten este enfoque pueden esperar beneficios considerables, como la eliminación de intermediarios, agilización de procesos, reducción de complejidad operativa y mejora en la transparencia y seguridad de las transacciones.

Desde el punto de vista teórico, la investigación aporta al campo de la integración de IoT y blockchain al proponer un método innovador que automatiza procesos contractuales mediante Smart Contracts. El valor teórico se centra en la optimización de la ejecución de contratos a través de la capitalización de la infraestructura de IoT, ofreciendo un enfoque específico y aplicable en entornos organizacionales. Si bien los resultados específicos pueden depender de la implementación y contexto, la propuesta teórica de integrar Smart Contracts

con IoT tiene el potencial de generalizarse a principios más amplios en la mejora de procesos contractuales interorganizacionales.

La investigación proporciona una metodología innovadora al introducir Smart Contracts que operan según agentes definidos en las plataformas IoT de cada organización. Esta metodología busca automatizar procesos contractuales, estableciendo condiciones y utilizando datos de IoT como activadores. La utilidad metodológica radica en la posibilidad de replicar este enfoque en diversos contextos organizacionales que buscan mejorar la eficiencia en la ejecución de contratos. Los principios metodológicos presentados en la investigación pueden guiar la implementación de soluciones similares en otras industrias o aplicaciones que busquen aprovechar las sinergias entre IoT y blockchain.

1.5 Contribuciones Principales

En el apartado de referencias se encuentran las bibliografías que sustentan la investigación y en esta sección se recopilan las publicaciones que están estrechamente vinculadas con los contenidos abordados en cada capítulo. Estas publicaciones abarcan congresos internacionales y revistas especializadas.

1.5.1. Publicaciones en Revistas

- Ericsson Mobility Report. (2022, Noviembre). *LTE Cat-1 devices are increasingly being used for a variety of use cases,* 11-15.

1.5.2. Publicaciones en Congresos

- Ali, J., & Ali, T. (2018). Towards Secure IoT Communication with Smart Contracts in a Blockchain Infrastructure.
- Kim, Y., Oh, H., & Kang, S. (2017, 06 5). Proof of Concept of Home IoT Connected Vehicles. *Sensors*.

- Christidis, K., & Devetsikiotis, M. (2016, 05 10). Blockchains and Smart Contracts for the Internet of Things.
- Islam, S. H., Pal, A. K., Samanta, D., & Bhattacharyya, S. (Eds.). (2022). *Blockchain Technology for Emerging Applications: A Comprehensive Approach*. Elsevier Science.

1.6 Estructura General de la Tesis

La estructura del trabajo está comprendida por IIX capítulos, los cuales describen los capítulos subsiguientes. Esta introducción sentará las bases para comprender el propósito y el alcance del trabajo de investigación.

1.6.1 Capítulos

- En el capítulo II - Trabajos Relacionados, se revisa la investigación y los trabajos previos relacionados con la intersección de Internet de las Cosas (IoT) y los Smart Contracts basados en tecnología Blockchain. Se exploran diferentes enfoques, casos de uso y aplicaciones prácticas en los que se utilizan estas tecnologías de manera conjunta. Se analizan los beneficios, desafíos y lecciones aprendidas de estos trabajos relacionados, proporcionando una base sólida para el desarrollo posterior del estudio.

- En el capítulo III - Conceptos de IoT, se explican las ideas centrales que subyacen a la Internet de las Cosas (IoT). Además se abordan elementos importantes del ecosistema IoT, como dispositivos, sensores, actuadores y conectividad. Se tratan los diseños y métodos de implementación más comunes, así como los protocolos de comunicación utilizados en el Internet de las Cosas. También se analizan los problemas de privacidad y seguridad de IoT.

- En el capítulo IV, Conceptos de Blockchain, se presentan los conceptos esenciales de la tecnología Blockchain. Se exploran los principios subyacentes de la cadena de bloques, incluyendo la descentralización, la inmutabilidad y la transparencia. Se describe la estructura básica de un bloque, el consenso distribuido y los algoritmos de minería utilizados para asegurar la red. También se examinan diferentes tipos de Blockchain, como las públicas y las privadas.
- En el capítulo V - Conceptos de Smart Contracts, se exploran los conceptos fundamentales de los Smart Contracts. Se define qué son los Smart Contracts y cómo funcionan. Se examina el papel de los Smart Contracts en la automatización de acuerdos y transacciones, eliminando intermediarios y aumentando la eficiencia. Se analizan los lenguajes de programación utilizados para desarrollar Smart Contracts, así como las plataformas y frameworks más populares. También se discuten las ventajas y los desafíos de los Smart Contracts en términos de seguridad y escalabilidad.
- En el capítulo VI - IoT y Smart Contracts, se explora la relación entre IoT y los Smart Contracts. Se analiza cómo la combinación de estas tecnologías puede ofrecer soluciones innovadoras en diversos campos, como la logística, la energía, la agricultura, entre otros. Se analizan casos de uso específicos donde IoT y los Smart Contracts se complementan mutuamente, aprovechando la conectividad y la automatización para crear sistemas más eficientes y seguros. Además, se analizan los desafíos y las consideraciones clave al implementar IoT y Smart Contracts de manera conjunta, incluyendo la escalabilidad, la interoperabilidad y la privacidad de los datos.
- En el capítulo VII - Prototipo: se implementa un prototipo abarcando un modelo de máquina de estado y diagrama de secuencia, se elabora un Smart Contract del caso de uso de agente de aparcamiento por detección de movimiento con el lenguaje de programación Solidity en ecosistemas IoT.

- En el capítulo VIII - Corroboración empírica: Se presenta la herramienta Remix IDE. Metamask. Se realizan pruebas simuladas en para las redes blockchain Ethereum. Se analizan recomendaciones.

1.6.2 Anexos

- Anexo I: Se presenta una breve historia del IoT. Bitcoin. Se presenta una breve historia de Blockchain. Breve historia de Ethereum y Smart Contract. Web 3.0 en el contexto de blockchain. Aplicaciones más usuales en dApp (DeFi, Gamificación, NFT). Blockchain Developer.
- Anexo II: Trabajo de investigación aceptado en el CoNaIISI 2023. Presentado el 2 de Noviembre del 2023 en la categoría Profesional/Investigador.

1.6.3 Acrónimos

1.6.4 Referencias

Capítulo 2 Trabajos Relacionados

2.1 Introducción

La gestión de flotas de vehículos ha sido un desafío constante para las industrias y organizaciones que dependen de una logística eficiente y rentable en la región. En los últimos años, la implementación de tecnologías de IoT ha permitido la integración de servicios y sensores en los vehículos, generando un amplio conjunto de datos que puede ser utilizado para mejorar la gestión de flotas.

Este capítulo describe investigaciones y enfoques previos relacionados con la gestión de flotas a través de la implementación de contratos inteligentes en vehículos con servicios IoT. Se revisan trabajos relacionados con la automatización y la gestión transparente de los servicios de IoT de vehículos a través de Smart Contracts y Blockchain.

Si bien en la región no hay proyectos similares, en otros países se puede encontrar trabajos que solucionen problemáticas similares, el objetivo es comprender los enfoques existentes, identificar fortalezas, limitaciones y la brecha de conocimiento, sentar las bases para proponer métodos innovadores de gestión de flotas. Esta revisión ayuda a justificar la singularidad y relevancia de la solución propuesta.

2.2 Sistema de Gestión de Transporte Inteligente

Si bien no se han encontrado proyectos e investigaciones en la región, se analizan proyectos similares implementados en diferentes partes del mundo y explorando sus beneficios y desafíos.

2.2.1 Blockchain Technology for Intelligent Transportation Systems: A Systematic Literature Review

En este artículo se revisa sistemáticamente la aplicación de blockchain en los sistemas de transporte inteligentes en general y en el Internet de Vehículos, de ahora en adelante IoV, en particular. Se estructura en cuatro partes principales: introducción a la tecnología blockchain, evolución de blockchain, estado del arte de las soluciones de IoV basadas en blockchain y resumen de problemas abiertos y futuras direcciones de investigación en BIoV. Se puede destacar que proporciona una visión general más amplia y completa de la investigación existente en la aplicación de blockchain en el ámbito del transporte inteligente y las redes IoV. Además, se destaca la importancia de clasificar las contribuciones de investigación según las capas de IoV y cómo esto puede ayudar a comprender mejor la aplicación de blockchain en el contexto específico del transporte inteligente. Sin embargo, sería beneficioso analizar críticamente si esta revisión consideró suficientemente los desafíos y limitaciones de la implementación de blockchain en el campo del transporte inteligente. Además, sería relevante explorar si se abordaron aspectos prácticos y operativos, como el escalado de blockchain en sistemas de transporte a gran escala y la interoperabilidad con tecnologías existentes (JABBAR et al., 2022).

Si bien el enfoque y la estructura es diferente, ya que proporcionado amplía la perspectiva para incluir un análisis más completo de la aplicación de blockchain en sistemas de transporte inteligentes y el IoV. En el contexto de sistemas de transporte y vehículos inteligentes, son útiles para obtener beneficios específicos de implementar blockchain en la gestión de datos en sistemas IoT.

2.2.2 Blockchain-Enabled Vehicular Ad Hoc Networks: A Systematic Literature Review

El artículo hace una revisión sistemática, se seleccionaron 68 estudios para proporcionar una visión general completa de blockchain y los contratos inteligentes en las redes vehiculares ad hoc (VANETs). Además, se propone un modelo de comunicación descentralizado para la implementación avanzada de blockchain en VANETs. Se hace hincapié en la identificación de problemas abiertos y en la exploración de las aplicaciones de blockchain en el contexto del IoV para cubrir brechas de investigación en redes de comunicación avanzadas en el Internet de las Cosas (Saad et al., 2022). Explorar las aplicaciones blockchain en el ámbito de los vehículos y las redes de comunicación y potenciar la blockchain para mejorar la seguridad, la descentralización y la eficiencia en estos contextos es útil para la investigación.

2.2.3 Proof of Concept of Home IoT Connected Vehicles

En este artículo, presenta el concepto de un vehículo conectado a IoT en el hogar con un asistente personal virtual basado en voz, compuesto por un agente de vehículo y un agente de hogar. Se evalúa este concepto implementando un teléfono inteligente vinculado a dispositivos IoT del hogar, conectados a un sistema de infoentretenimiento en el vehículo. También se utiliza un dispositivo de entrada basado en lenguaje natural en el teléfono inteligente y dispositivos IoT del hogar basados en la nube. Los escenarios de servicios conectados entre el hogar y el vehículo que buscan reducir las tareas simples y repetitivas para mejorar la eficiencia de la movilidad urbana en entornos de IoT se respaldan mediante análisis de pruebas reales en vehículos e investigación sobre el estilo de vida. Se obtienen beneficios significativos al unificar tareas rutinarias repetitivas en una sola tarea ejecutada mediante un

comando y al ejecutar tareas esenciales de forma automática sin necesidad de solicitud (Kim et al., 2017).

2.3 Smart Contracts aplicado al IoT

2.3.1 Blockchains and Smart Contracts for the Internet of Things

En este artículo se presenta una investigación que explora la posibilidad de utilizar Blockchains en el sector de IoT. Se analiza cómo las Blockchains permiten una red distribuida peer-to-peer donde miembros no confiables pueden interactuar entre sí sin la necesidad de un intermediario de confianza, de una manera verificable. Además, se examinan los Smart Contracts que residen en la Blockchain y que permiten la automatización de procesos multi-etapa. Luego, se explora cómo la combinación de Blockchain e IoT facilita el intercambio de servicios y recursos, creando un mercado de servicios entre dispositivos, y cómo permite la automatización de flujos de trabajo existentes, reduciendo tiempos y costos (Christidis & Devetsikiotis, 2016).

Se enfatiza la potencia de la combinación de blockchain e IoT. Las blockchain ofrecen sistemas distribuidos peer-to-peer resistentes, que permiten interactuar con otros participantes de forma confiable y auditable. Los smart contracts permiten la automatización de procesos complejos. Al combinar estos elementos con los dispositivos IoT, se logra la automatización de flujos de trabajo de manera única y criptográficamente verificable, lo que resulta en ahorros significativos en tiempo y costos (Christidis & Devetsikiotis, 2016).

2.4 Implementaciones con Solidity en el entorno IoT

2.4.1 Building trust among things in omniscient Internet using Blockchain Technology (Proyecto IoTeX)

Se reconocen los desafíos en términos de seguridad y privacidad de datos que surgen con el crecimiento constante del número de dispositivos de IoT. Se enfatiza la necesidad de abordar estos desafíos en un ecosistema vertical donde los dispositivos tienen recursos limitados y no pueden implementar patrones de seguridad complejos (Boncea et al., 2019).

Por otro lado, el artículo sugiere el uso de un libro de contabilidad digital descentralizado y tecnologías de blockchain para establecer confianza en dispositivos de la capa de puerta de enlace y computación en la nube y reconoce que una Blockchain completamente descentralizada puede tener una capacidad de procesamiento baja, lo cual es inaceptable para implementaciones de IoT a gran escala. Para abordar esto, sugieren sacrificar parte de la descentralización y adoptar una arquitectura de cadenas de bloques dentro de cadenas de bloques, donde la carga de trabajo se realiza en clústeres locales con alta autonomía y formas estandarizadas de comunicación con la red de subcadenas (Boncea et al., 2019).

El artículo menciona proyectos como IoTeX y Hyperledger que se enfocan en soluciones para implementaciones de IoT. IoTeX es una plataforma de IoT más rápida y flexible que admite la Máquina Virtual Ethereum y Solidity, mientras que Hyperledger es un incubador de proyectos de cadenas de bloques que ofrece una amplia gama de proyectos con un enfoque en la reutilización de componentes comunes y la innovación rápida en tecnologías de contabilidad distribuida (Boncea et al., 2019).

2.4.2 Proyecto Motoro Coin

https://github.com/blockchain-IoT/Motoro

El proyecto se centra en el desarrollo de un sistema de alquiler de vehículos basado en la tecnología blockchain y el IoT. El objetivo principal es crear una plataforma descentralizada que permita a los usuarios alquilar vehículos de forma segura y transparente, eliminando intermediarios y reduciendo los costos asociados. Utiliza contratos inteligentes en la cadena de bloques Ethereum para establecer y hacer cumplir los términos de alquiler entre los propietarios de vehículos y los arrendatarios. Estos contratos automatizan el proceso de alquiler, incluyendo la verificación de identidad, el pago y la gestión de los aspectos logísticos (Mikolajczyk & Ordaz, n.d.).

Además, se implementan soluciones de IoT para mejorar la experiencia de alquiler. Los vehículos están equipados con dispositivos de seguimiento y sensores que recopilan datos en tiempo real, como la ubicación y el estado del vehículo. Estos datos se registran en la cadena de bloques para proporcionar un historial completo y transparente de la utilización del vehículo (Mikolajczyk & Ordaz, n.d.).

El proyecto también aborda desafíos específicos del sector, como el robo de vehículos y el mantenimiento adecuado. La tecnología blockchain permite rastrear y verificar la propiedad de los vehículos, lo que ayuda a prevenir el robo y proporciona un nivel adicional de seguridad. Además, los contratos inteligentes pueden incluir cláusulas relacionadas con el mantenimiento y la reparación de los vehículos, asegurando que estén en buenas condiciones para su uso (Mikolajczyk & Ordaz, n.d.).

2.4.3 Proyecto Full DIY (R, 2022)

Este proyecto miniatura se enfoca en soluciones para hogares inteligentes que permiten controlar dispositivos eléctricos como luces, ventiladores, calentadores de agua,

bombas de agua, accesorios de cocina, entre otros. Los contratos inteligentes funcionan en la tecnología blockchain y controlan estos dispositivos. En este proyecto, se construye una red P2P con dos VM basadas en GCP y un sistema basado en Raspberry Pi. Las dos VM se utilizan como mineros y aprueban las transacciones desde el minero 1 hacia el Raspberry Pi. El Raspberry Pi está conectado a un relé que enciende y apaga una luz. Un contrato inteligente se despliega en la blockchain a través de Web3.0 y controla la luz mediante transacciones como depositar y retirar tokens. Este proyecto miniatura involucra tecnologías como blockchain, IoT, computación en la nube y Web3.0 (R, 2022).

Especifica los siguientes entornos de desarrollo son necesarios:

- Máquinas virtuales de Google Cloud (2 unidades) para ejecutar los mineros (JavaScript).
- Raspberry Pi como un nodo de blockchain (JavaScript).
- Entorno Truffle para construir el contrato inteligente (Solidity).
- Controlar la luz desde el minero 1 a través del Raspberry Pi (JavaScript).

2.4.4 Proyecto IDMoB: IoT Data Marketplace on Blockchain

En este proyecto comieza inicialmente que los objetivos de las investigaciones anteriores fueron explorar formas de integrar dispositivos IoT de bajo consumo a una infraestructura basada en blockchain, así como crear un backend descentralizado para el almacenamiento de datos. En este artículo, se busca extender ese objetivo a un mercado de datos más amplio que involucre a múltiples partes, centrándose en aplicaciones de IoT no críticas y no en tiempo real. El objetivo es crear una plataforma descentralizada y sin confianza para almacenar y acceder a datos de IoT, lo cual impactará positivamente a los fabricantes de dispositivos IoT, proveedores de IA/ML y los usuarios finales. Se espera que dicho mercado democratice el acceso a datos consensuados y aumente la calidad y variedad

de servicios ofrecidos, lo cual beneficiará a los usuarios. Con esta idea, se elabora una implementación de un mercado de datos como prueba de concepto con Smart Contracts en la plataforma Ethereum, utilizando Swarm como sistema de almacenamiento. Este mercado proporciona un mecanismo de consulta flexible para los consumidores de datos y contiene un mecanismo de votación para eliminar proveedores de datos no confiables (Ozyilmaz et al., 2018). El código del Smart Contract está disponible como código abierto en GitHub bajo el nombre "IDMoB: IoT Data Marketplace on Blockchain".

2.4.5 A Prototype of Supply Chain Traceability using Solana as blockchain and IoT

En este artículo se presenta un prototipo que integra blockchain e IoT para digitalizar la información a lo largo de una cadena de suministro genérica. El objetivo principal es mejorar la trazabilidad y confiabilidad de la cadena de suministro al implementar contratos inteligentes en una cadena segura y autenticada (Ashraf & Heavey, 2023). Blockchain es una base de datos descentralizada que brinda seguridad en el registro de información y la trazabilidad basada en blockchain puede abordar deficiencias existentes en soluciones centralizadas.

El artículo presenta una arquitectura de software que combina blockchain con dispositivos IoT para permitir la trazabilidad de un producto genérico desde su origen hasta su destino final, pasando por proveedores de múltiples niveles, logística, fabricantes y clientes. La arquitectura de software utiliza la blockchain Solana para implementar los procesos y la lógica empresarial de la cadena de suministro. Esta blockchain fue seleccionada después de revisar varias redes blockchain ampliamente utilizadas, principalmente debido a su velocidad y costo de transacciones. Dentro de la blockchain se almacenan los datos y eventos relacionados con la cadena de suministro que se comunican a través de canales de aplicación

móvil e internet, utilizando las bibliotecas nativas de la blockchain Solana (Ashraf & Heavey, 2023). Los dispositivos IoT utilizados incluyen una pasarela en la nube Sigfox y Sensit, que utiliza una red de telecomunicaciones de área amplia y baja potencia (LPWAN, por sus siglas en inglés) para la transferencia de datos. En el dispositivo IoT se almacenan datos como temperatura, humedad, luz, ubicación, inclinación, apertura de puertas, vibración y campo magnético.

El objetivo final es utilizar tecnologías existentes para desarrollar una arquitectura de software que pueda implementarse como una blockchain genérica para cadenas de suministro (Ashraf & Heavey, 2023). Esta solución busca mejorar la trazabilidad y confiabilidad de las cadenas de suministro mediante la integración de blockchain e IoT.

2.5 Críticas

Si bien es indudable la intención de los proyectos de IoT de brindar soluciones y control a través de tecnologías como Blockchain y Smart Contracts, es crucial analizar de manera crítica algunos desafíos que estos proyectos enfrentan.

En primer lugar, es importante destacar los altos costos asociados a la implementación de estos proyectos. Establecer un plan de acción para ejecutar proyectos de IoT basados en Blockchain y Smart Contracts resulta sumamente costoso en la actualidad. Cada proyecto debe evaluar cuidadosamente la relación entre los costos incurridos y los beneficios que se obtendrán a largo plazo. Esta evaluación de costo-beneficio es fundamental para garantizar la viabilidad económica de dichos proyectos.

Otro desafío significativo radica en la falta de estandarización de los contratos inteligentes en el ámbito del IoT. Esta falta de estandarización implica que cada empresa debe crear sus propias versiones de contratos inteligentes, lo cual genera confusión y complejidad cuando diferentes dispositivos de diferentes fabricantes intentan interactuar entre sí. La falta

de un estándar común dificulta la interoperabilidad y limita la facilidad de uso en el ecosistema del IoT. Es esencial abordar este problema mediante la colaboración y el desarrollo de estándares abiertos que promuevan la compatibilidad y la cohesión en el ecosistema del IoT.

Además, es crucial considerar los posibles desafíos regulatorios y legales asociados con los contratos inteligentes en el IoT. Dada la novedad de esta tecnología, el estatus regulatorio de los contratos inteligentes a menudo es incierto y puede variar según las jurisdicciones. Esto plantea preocupaciones en cuanto a la validez legal y la resolución de disputas en el contexto de los contratos inteligentes. Es necesario abordar estas incertidumbres regulatorias y legales para fomentar la confianza y la adopción generalizada de los contratos inteligentes en el ámbito del IoT.

Capítulo 3 Internet of Things

3.1 Introducción

En este capítulo se explican las principales características, desafíos y aplicaciones de la IoT, como también las ideas centrales del concepto de IoT. Se abordan temas importantes dentro del ecosistema IoT, se conceptualizan los dispositivos, sensores, actuadores más relevantes y la conectividad. Se muestran los diseños y métodos de implementación más comunes, así como los protocolos de comunicación utilizados en el Internet de las Cosas. También se analizan los problemas de privacidad y seguridad de IoT.

La Internet de las Cosas, comúnmente abreviada como IoT, es una revolucionaria convergencia tecnológica que está transformando la forma en que interactuamos con el mundo que nos rodea. En esencia, la IoT se refiere a la interconexión de objetos cotidianos a través de Internet, permitiendo que estos objetos recopilen y compartan datos de manera

autónoma. Desde electrodomésticos inteligentes y sistemas de seguridad hasta vehículos conectados y dispositivos médicos, la IoT se ha convertido en una red invisible que enlaza los objetos físicos con el mundo digital.

Esta interconexión masiva de dispositivos y sensores proporciona una riqueza de información en tiempo real que es valiosa en una amplia gama de aplicaciones. Ya sea para mejorar la eficiencia en la industria, optimizar el uso de recursos, facilitar la toma de decisiones más informadas o habilitar la automatización de tareas cotidianas, la IoT ha llegado para revolucionar la forma en que vivimos y trabajamos.

3.2 Definición de Internet of Thing

La IoT es un concepto que se refiere a la interconexión de objetos físicos, dispositivos y sistemas a través de Internet, permitiéndoles recopilar, compartir y utilizar datos para mejorar la eficiencia, la toma de decisiones, la automatización y la comodidad en diversas aplicaciones. Un sistema de IoT representa una red inteligente que establece vínculos entre todos los objetos y la Internet, permitiendo el intercambio de información mediante protocolos concebidos por el Internet Engineering Task Force (IETF). Como resultado de esto, se posibilita el acceso a cualquier objeto en cualquier momento y lugar. Sensores minúsculos, incorporados en elementos cotidianos, constituyen el soporte esencial de una red de IoT. Un aspecto destacado radica en la ausencia de necesidad de intervención humana en las interacciones de los dispositivos IoT. Esta tecnología no solo reduce significativamente el esfuerzo humano, sino que también optimiza la utilización de recursos, permite un monitoreo y seguimiento en tiempo real, mejora la recolección de datos y optimiza el uso del tiempo. Ejemplos de su aplicación abarcan desde dispositivos portátiles hasta la agricultura inteligente, venta minorista inteligente y la monitorización de la salud. A partir de la singularidad de sus técnicas de direccionamiento, el IoT posibilita la creación de nuevas

aplicaciones y servicios que interactúan con diversos objetos y elementos (Paricherla et al., 2022).

3.2.1 IoV

En consonancia con el progreso de la industria, se observa un crecimiento acelerado en la cantidad de vehículos en circulación. Este aumento de vehículos da lugar a inquietudes en cuanto a la seguridad, lo que motiva la necesidad de establecer un medio de comunicación seguro. El IoV emerge como protagonista en el contexto de la industria 4.0. Es innegable que el IoV proyecta un futuro prometedor y beneficioso, al brindar mejoras en la seguridad vial, disminuir los impactos ambientales, optimizar la utilización del espacio y ejercer control sobre los costos asociados. El ecosistema del IoV abarca tanto componentes de hardware como software, acompañados por servicios diversos y una amplia variedad de tecnologías de red que van desde Bluetooth y redes celulares hasta Wi-Fi y tecnología 5G. Asimismo, se emplean diversos tipos de comunicación, como la vehículo a vehículo (V2V), vehículo a infraestructura (V2I), vehículo a red (V2N) y vehículo a peatón (V2P), para garantizar la conectividad y la interacción fluida en este entorno dinámico. El IoV no solo implica la interacción entre vehículos, sino también con el entorno circundante, lo cual permite el acceso a datos esenciales en tiempo real para la seguridad y la eficiencia del tráfico. Para garantizar la integridad de la red IoV y prevenir posibles situaciones peligrosas causadas por la manipulación de datos maliciosos, resulta crucial el desarrollo de un sólido marco de autenticación. Dicho marco debe tener la capacidad de verificar la autenticidad de los datos en cuestión de milisegundos, asegurando así la confiabilidad y la seguridad en las decisiones basadas en la información de la red IoV (Kumar Sadhu et al., 2022).

3.3 Dispositivos IoT

Los dispositivos de la IoT son objetos y sistemas físicos que han sido equipados con capacidades de conectividad a Internet y sensores para recopilar datos, actuadores para realizar acciones, y controladores para gestionar y automatizar procesos.

- **Sensores**: Estos dispositivos detectan y recopilan información del entorno, como temperatura, movimiento, humedad, o calidad del aire.
- **Actuadores**: Los actuadores permiten a los dispositivos IoT llevar a cabo acciones en función de los datos recopilados. Esto podría incluir ajustar la temperatura, bloquear una puerta, o activar un sistema de seguridad.
- **Controladores**: Los controladores, que pueden ser aplicaciones móviles o plataformas en la nube, permiten a los usuarios gestionar y supervisar dispositivos IoT, así como automatizar procesos y tomar decisiones basadas en datos.

3.3.1 Sensores

1. **Sensores de Calidad del Aire:** Detectan la calidad del aire y los niveles de contaminantes en interiores y exteriores, utilizados en sistemas de gestión de calidad del aire y salud ambiental.
2. **Sensores de Movimiento:** Detectan movimiento y actividad, utilizados en sistemas de seguridad, iluminación automatizada y seguimiento de ocupación en edificios.
3. **Medidores Inteligentes de Energía:** Monitorean y registran el consumo de energía eléctrica en tiempo real, permitiendo la gestión eficiente de la energía en hogares y empresas.

3.3.2 Actuadores

1. **Termostatos Inteligentes:** Controlan la temperatura ambiente y permiten la automatización del sistema de calefacción o aire acondicionado en hogares y edificios.
1. **Cerraduras Inteligentes:** Permiten el control de puertas y cerraduras de forma remota a través de una aplicación móvil, brindando seguridad y flexibilidad en el acceso.
2. **Vehículos Conectados:** Incorporan tecnología IoT para ofrecer servicios de navegación en tiempo real, diagnósticos de vehículos y entretenimiento a bordo.

3.3.3 Controladores

1. **Asistentes de Voz:** Actúan como controladores para otros dispositivos IoT mediante comandos de voz, permitiendo la automatización del hogar y la interacción con servicios en línea.
2. **Aplicaciones Móviles para el Control de Dispositivos:** Permiten a los usuarios controlar y gestionar una variedad de dispositivos IoT desde sus dispositivos móviles, como ajustar luces, termostatos o cámaras de seguridad.
3. **Plataformas de Gestión de IoT en la Nube:** Ofrecen una gestión centralizada para miles de dispositivos IoT, permitiendo el monitoreo, la recopilación de datos y la toma de decisiones basada en datos en escala empresarial.

3.4 Métodos de implementación y conectividad

En el emocionante mundo de la IoT, la implementación y el diseño son elementos cruciales que determinan el éxito y la eficiencia de los proyectos. Las metodologías de implementación y diseño son enfoques estructurados que guían el desarrollo de sistemas IoT, asegurando que estos sean efectivos, seguros y escalables.

La implementación se refiere a la creación física y operativa de dispositivos IoT, desde la selección de componentes y sensores hasta la programación de software y la integración en una red. Por otro lado, el diseño se concentra en la arquitectura general del sistema, considerando aspectos como la conectividad, la seguridad, la eficiencia energética y la gestión de datos.

Estas metodologías son vitales debido a la diversidad de aplicaciones de la IoT, desde ciudades inteligentes y salud digital hasta la gestión de la cadena de suministro y la agricultura de precisión. Además, la IoT presenta desafíos únicos, como la gestión de grandes volúmenes de datos, la seguridad de dispositivos conectados y la optimización del rendimiento.

Se abordan implementaciones y protocolos de conectividad más comunes que existen en las plataformas IoT:

1. **Arquitectura Cliente-Servidor:** En este enfoque, los dispositivos IoT actúan como clientes que envían datos a servidores en la nube para su procesamiento y almacenamiento. Esto permite una fácil gestión y escalabilidad, pero a menudo implica latencia y dependencia de la conectividad a Internet.
2. **Edge Computing:** Los dispositivos IoT pueden realizar cierto procesamiento y análisis de datos en el borde (edge) de la red, antes de enviar datos a la nube. Esto reduce la latencia y puede ser crítico en aplicaciones donde el tiempo es esencial.
3. **Fog Computing:** Similar a Edge Computing, pero en este caso, los datos se procesan en puntos intermedios de la red, conocidos como "nodos de niebla". Esto es útil para reducir la carga en la nube y mejorar la eficiencia en aplicaciones distribuidas.
4. **Malla de Dispositivos (Mesh Networks):** En aplicaciones de IoT donde la conectividad es crítica, como en redes de sensores, los dispositivos pueden

comunicarse directamente entre sí y retransmitir datos a lo largo de la red en lugar de depender de una conexión central. Esto mejora la resiliencia y la cobertura.

5. **Arquitectura de 3 Capas:** En aplicaciones industriales, se utiliza una arquitectura de 3 capas que incluye dispositivos en el campo, pasarelas intermedias y servidores de gestión en la nube. Esto permite una comunicación eficiente y segura.

3.4.1 Protocolos de Comunicación

1. **MQTT (Message Queuing Telemetry Transport):** Es un protocolo ligero de publicación/suscripción que se utiliza para la transmisión de mensajes en tiempo real. Es ampliamente utilizado en aplicaciones de IoT y M2M (máquina a máquina).
2. **HTTP/HTTPS:** Protocolos web estándar utilizados para la comunicación entre dispositivos IoT y servidores en la nube. HTTP es utilizado para aplicaciones menos críticas, mientras que HTTPS proporciona seguridad adicional.
3. **CoAP (Constrained Application Protocol):** Un protocolo de aplicación web especialmente diseñado para dispositivos IoT con recursos limitados, como sensores y actuadores.
4. **AMQP (Advanced Message Queuing Protocol):** Un protocolo de mensajería que se utiliza para aplicaciones de IoT que requieren alta velocidad y eficiencia en la comunicación.
5. **LoRaWAN (Long Range Wide Area Network):** Un protocolo de comunicación inalámbrica de largo alcance que se utiliza en aplicaciones de IoT de bajo consumo de energía y larga distancia, como el seguimiento de activos.
6. **Zigbee y Z-Wave:** Protocolos de comunicación inalámbrica utilizados en aplicaciones de automatización del hogar y edificios inteligentes.

7. **Bluetooth y Bluetooth LE (Low Energy):** Utilizados en dispositivos IoT de corto alcance, como wearables y dispositivos de salud.
8. **NFC (Near Field Communication):** Se utiliza en aplicaciones de corto alcance, como el pago sin contacto y la identificación.

3.4.2 Concepto de señales

En el contexto de la teoría de señales, una "señal" se define como cualquier cantidad física que varía con respecto al tiempo, el espacio u otras variables independientes y que puede transmitir información. Una señal puede representar fenómenos como sonido, luz, temperatura, posición, entre otros.

Las "señales sin procesar" en este contexto se refieren a la representación original e inalterada de la información medida o capturada por un sensor o dispositivo de medición. Estas señales son la salida directa del sensor antes de pasar por algún tipo de procesamiento o tratamiento. Representan la información cruda tal como fue registrada, sin manipulación o análisis adicional. La manipulación de estas señales sin procesar generalmente ocurre en etapas posteriores para extraer información significativa o realizar ajustes necesarios para aplicaciones específicas.

3.5 Seguridad y privacidad

La privacidad y la seguridad son cuestiones críticas en los sistemas IoT debido a la creciente cantidad de datos personales y sensibles que se recopilan y transmiten a través de estos dispositivos interconectados. Acá hay algunas consideraciones importantes sobre privacidad y seguridad en las implementaciones de IoT:

Privacidad en IoT:

1. **Recopilación de Datos Sensibles:** Los dispositivos IoT a menudo recopilan datos personales y sensibles, como ubicaciones, patrones de comportamiento y datos de salud. Es esencial garantizar que estos datos se manejen de manera ética y se respeten las regulaciones de privacidad.
2. **Consentimiento del Usuario:** Los usuarios deben ser informados y dar su consentimiento antes de que se recopilen sus datos. Las políticas de privacidad transparentes y las opciones de exclusión voluntaria son esenciales.
3. **Encriptación de Datos:** Los datos transmitidos entre dispositivos y servidores deben estar encriptados para protegerlos de posibles amenazas. El cifrado garantiza que los datos no sean accesibles para terceros no autorizados.
4. **Almacenamiento Seguro:** Los datos almacenados en servidores o en dispositivos deben ser protegidos de accesos no autorizados. Esto implica la implementación de medidas de seguridad, como autenticación y acceso basado en roles.

Seguridad en IoT:

1. **Actualización de Firmware:** Los dispositivos IoT deben recibir actualizaciones regulares de firmware para corregir vulnerabilidades conocidas. Los fabricantes deben facilitar la actualización de dispositivos.
2. **Autenticación y Autorización:** Los dispositivos y usuarios deben autenticarse y autorizarse adecuadamente antes de acceder a sistemas o datos. La autenticación de dos factores es una medida de seguridad efectiva.
3. **Protección contra Ataques de Denegación de Servicio (DDoS):** Los dispositivos IoT pueden ser blanco de ataques DDoS que interrumpan su funcionamiento. La mitigación de DDoS es esencial para garantizar la disponibilidad.

4. **Gestión de Claves y Certificados:** Las claves y los certificados de seguridad deben manejarse adecuadamente para evitar su exposición y garantizar la autenticación segura.
5. **Segmentación de Redes:** La segmentación de redes permite aislar dispositivos en redes separadas, reduciendo la superficie de ataque.
6. **Auditoría y Monitoreo Continuo:** La implementación de sistemas de auditoría y monitoreo constante permite detectar actividades inusuales o potenciales amenazas de seguridad.
7. **Educación y Concienciación:** Los usuarios y operadores de dispositivos IoT deben estar capacitados sobre las mejores prácticas de seguridad y cómo proteger sus sistemas.

3.6 Plataforma IoT

Una plataforma IoT es un conjunto de tecnologías y herramientas que permiten el desarrollo, la implementación y la gestión de aplicaciones y dispositivos IoT. Las aplicaciones del IoT pueden ser empleadas de diversas formas para apoyar a sistemas y negocios en la simplificación, mejora, automatización y control de procesos. El IoT también puede ser utilizado para proporcionar datos importantes, el rendimiento de actividades, e incluso factores ambientales que necesitan ser monitoreados de manera continua y remota. Por lo tanto, las aplicaciones del IoT pueden contribuir en la creación de nuevos sistemas y estrategias empresariales, al mismo tiempo que brindan a las empresas los datos instantáneos que necesitan para desarrollar productos y servicios de manera eficiente (Kumar Sadhu et al., 2022).

A continuación se describen algunos elementos clave de una plataforma IoT:

1. **Conectividad:** Las plataformas IoT proporcionan la capacidad de conectar y comunicar una variedad de dispositivos y sensores a través de múltiples protocolos de comunicación, como MQTT, CoAP, HTTP y más.
2. **Gestión de Dispositivos:** Permiten la gestión centralizada de dispositivos IoT, lo que incluye la configuración, actualización de firmware, supervisión del estado y la seguridad.
3. **Recopilación y Almacenamiento de Datos:** Facilitan la recopilación de datos de dispositivos y sensores, así como su almacenamiento en bases de datos o sistemas de gestión de datos.
4. **Análisis de Datos:** Ofrecen capacidades para analizar datos en tiempo real, identificar patrones, detectar anomalías y generar informes para la toma de decisiones.
5. **Seguridad:** Incluyen medidas de seguridad para proteger la integridad y la confidencialidad de los datos transmitidos y almacenados, además de garantizar la autenticación y la autorización seguras.
6. **Integración con Aplicaciones:** Las plataformas IoT suelen ofrecer API (Interfaz de Programación de Aplicaciones) para integrarse con otras aplicaciones empresariales, sistemas de automatización y servicios en la nube.
7. **Escalabilidad:** Deben ser capaces de manejar un crecimiento significativo en el número de dispositivos y la cantidad de datos generados a medida que se expande la implementación IoT.
8. **Diseño de Reglas (Agentes) y Automatización:** Permiten la creación de reglas y lógica empresarial que facilitan la automatización de tareas y respuestas a eventos específicos.
9. **Interfaz de Usuario:** Ofrecen una interfaz de usuario para la supervisión y el control de dispositivos y datos, a menudo a través de aplicaciones web o móviles.

10. **Gestión de Costos y Eficiencia Energética:** Pueden proporcionar herramientas para gestionar el consumo de energía y optimizar el uso de recursos, lo que es crítico en aplicaciones de IoT.

3.7 Resumen

En este capítulo, se ha explorado los conceptos fundamentales de la IoT. Comenzando comprendiendo que la IoT es la interconexión de objetos físicos a través de Internet, lo que permite la recopilación y el intercambio de datos para mejorar una amplia gama de aplicaciones. Luego, se abordó los métodos de implementación, que son esenciales para dar vida a los dispositivos IoT. Desde arquitecturas cliente-servidor y Edge Computing hasta redes de malla y sistemas de automatización, estos métodos proporcionan estructura y eficiencia en la creación de sistemas IoT.

La conectividad es un elemento crucial en la IoT, y se exploraron protocolos como MQTT, HTTP, y Zigbee, que permiten la comunicación entre dispositivos y sistemas. Estos protocolos son la base para una conectividad efectiva y segura en el mundo IoT. La privacidad y la seguridad también fueron temas destacados. La IoT implica la recopilación de datos sensibles y personales, por lo que es esencial garantizar la privacidad del usuario y proteger los sistemas contra amenazas. Se consideraron cuestiones como el consentimiento del usuario, la encriptación de datos y la seguridad de los dispositivos. Finalmente, se abordó el concepto de plataformas IoT, que son herramientas vitales en el desarrollo y la gestión de sistemas IoT. Desde la conectividad hasta la analítica de datos y la seguridad, estas plataformas ofrecen una infraestructura sólida para implementar aplicaciones IoT exitosas.

Capítulo 4 Blockchain

4.1 Introducción

La tecnología Blockchain se ha convertido en una de las innovaciones más impactantes de la última década, con un potencial revolucionario que se extiende mucho más allá de su asociación inicial con las criptomonedas. La historia de esta tecnología, se puede encontrar en el Anexo I. En este capítulo, se explora en profundidad los conceptos clave que sobre esta tecnología disruptiva y su influencia en diversos sectores.

Se define qué es Blockchain y cómo funciona, destacando su capacidad para crear registros digitales seguros e inmutables. Además, se examinan los conceptos subyacentes esenciales, como la descentralización, la distribución y la transparencia, que son los pilares de la tecnología Blockchain.

Luego, se explica la estructura básica de un bloque, desglosando cómo las transacciones se almacenan y protegen de manera segura. En este contexto, se explora cómo se logra la inmutabilidad de los datos, garantizando que la información registrada no pueda ser alterada o eliminada.

Uno de los aspectos más críticos de Blockchain es su capacidad para llegar a consensos en una red descentralizada. Se analizan varios algoritmos de consenso, desde el Proof of Work (PoW) hasta el Practical Byzantine Fault Tolerance (PBFT), y cómo influyen en la seguridad y el rendimiento de las redes Blockchain.

Además, se abordan la variedad de tipos de Blockchain, incluyendo redes privadas y públicas, y sus respectivas aplicaciones en diversos contextos.

4.2 Definición de Blockchain

La blockchain es un sistema digital descentralizado y distribuido que consiste en una cadena de bloques que contienen datos de transacciones que pueden compartirse entre los participantes conectados a la red de forma encriptada (Islam et al., 2022). Cada bloque contiene un conjunto de transacciones y una referencia al bloque anterior, creando una cadena continua de datos que no se puede modificar. La característica más distintiva de Blockchain es su capacidad para asegurar que los datos permanezcan inalterables y transparentes, lo que lo hace valioso en una amplia variedad de aplicaciones, desde criptomonedas como Bitcoin hasta la gestión de contratos y el seguimiento de activos. La seguridad y la resistencia a la manipulación son aspectos fundamentales de esta tecnología, ya que los datos registrados no pueden ser modificados sin el consenso de la mayoría de los participantes en la red, lo que garantiza su confiabilidad y confianza.

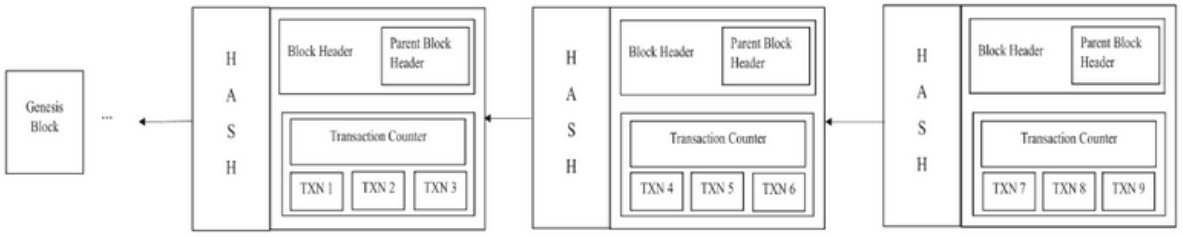
En la figura 4.1, cada bloque en esta cadena contiene un "hash" (un valor único y fijo) que se conoce como "parent block hash" (hash del bloque padre). Este valor contiene la identidad del bloque anterior en la secuencia. En otras palabras, cada bloque lleva consigo una referencia al bloque que lo precede. Esta referencia se logra a través del "hash" del bloque anterior (Islam et al., 2022).

Esta característica de los "parent block hashes" es fundamental en las cadenas de bloques, ya que proporciona un mecanismo para garantizar la integridad de la cadena. Si en algún momento se intenta eliminar, reemplazar o mover bloques en la cadena, estos cambios se pueden detectar fácilmente al verificar que los "parent block hashes" coinciden correctamente en la secuencia. Si los hashes no coinciden, se sabe que ha ocurrido una alteración en la cadena (Islam et al., 2022).

Cabe destacar que el primer bloque en una cadena de bloques se conoce como el "bloque génesis". Dado que no tiene un bloque previo al que hacer referencia, su "parent

block hash" se establece en un valor específico, generalmente 0. Esto indica que no hay un bloque anterior al bloque génesis, ya que marca el inicio de la cadena el encabezado del bloque (block header) y el cuerpo del bloque (block body) (Islam et al., 2022).

Figura 4.1: Blockchain



(Islam et al., 2022)

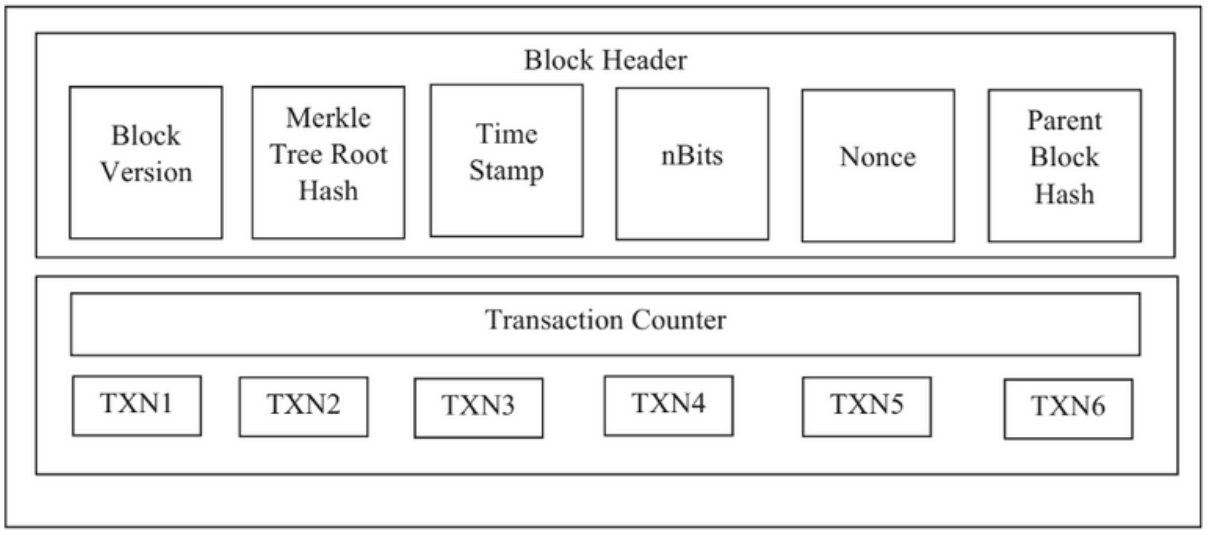
En la figura 4.2, se representa un bloque, cada bloque consta de dos partes principales (Islam et al., 2022):

- Encabezado del Bloque (Block Header): El encabezado del bloque contiene varios campos clave que son esenciales para el funcionamiento de la cadena de bloques:
- Versión del Bloque (Block version): Este campo indica qué conjunto de reglas de validación de bloque o protocolos se deben seguir para verificar la autenticidad del bloque.
- Hash de la Raíz del Árbol de Merkle (Merkle tree root hash): Este valor es el hash de todos los datos de las transacciones presentes en el cuerpo del bloque. Se utiliza para garantizar la integridad de las transacciones.
- Marca de Tiempo (Timestamp): Proporciona la fecha y hora actual en segundos desde el 1 de enero de 1970. Este valor se utiliza para registrar cuándo se creó el bloque.
- nBits: Este campo representa el umbral de dificultad requerido para que el hash del bloque sea considerado válido. En otras palabras, es el nivel de dificultad que debe alcanzarse al calcular el hash del bloque.

- Nonce: Es un campo de 4 bytes que comienza en 0 y se incrementa con cada cálculo de hash. Los mineros de criptomonedas ajusta este valor para encontrar un hash válido que cumpla con el umbral de dificultad requerido.
- Hash del Bloque Padre (Parent block hash): Como se mencionó anteriormente, este es un valor hash de 256 bits que apunta al bloque anterior en la cadena. Sirve para conectar los bloques en una secuencia continua.

El encabezado del bloque contiene información crucial para verificar y validar la autenticidad del bloque, asegurando que cumple con las reglas y protocolos establecidos en la red de la cadena de bloques.

Figura 4.2: Estructura de un bloque



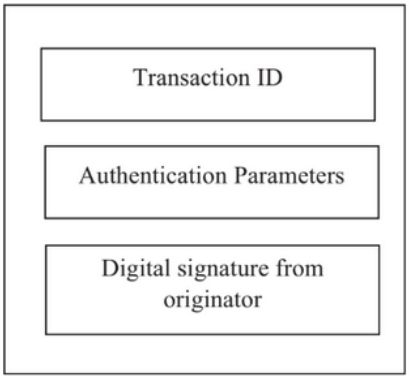
(Islam et al., 2022)

El cuerpo de un bloque consta de dos campos principales: el contador de transacciones (transaction counter) y las transacciones (TXN). La cantidad máxima de transacciones que un bloque puede contener puede variar dependiendo del tamaño de cada transacción y del tamaño total del bloque (Fig. 4.3) (Islam et al., 2022).

Para validar las transacciones, la cadena de bloques utiliza la criptografía asimétrica o sistemas de criptografía de clave pública. Estos mecanismos hacen uso de una firma digital para verificar la autenticidad de las transacciones. El algoritmo de firma digital típico

utilizado en la cadena de bloques es el algoritmo de firma digital de curva elíptica (Islam et al., 2022).

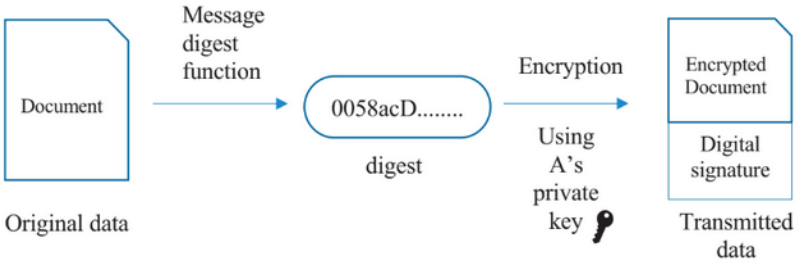
Figura 4.3: Transacción



(Islam et al., 2022)

Por ejemplo, cuando un usuario A desea enviar un mensaje o realizar una transacción a otro usuario B, se produce el siguiente proceso (Fig. 4.4). En este proceso, se utiliza una firma digital para garantizar que la transacción sea auténtica y que solo los participantes autorizados puedan realizar y validar la transacción en la cadena de bloques.

Figura 4.4: Firma



(Islam et al., 2022)

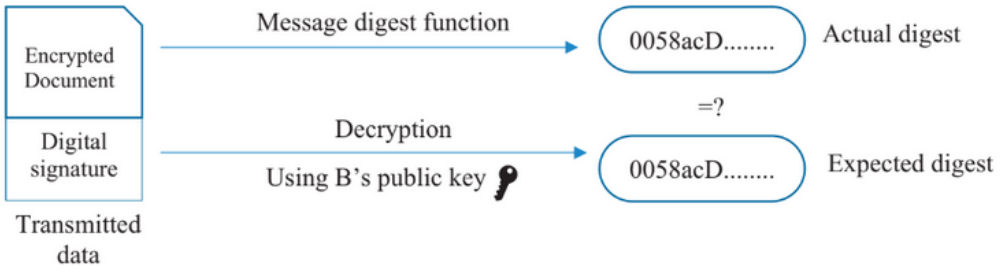
El proceso de firma y verificación de documentos en el contexto de la cadena de bloques utilizando criptografía de clave pública:

- **Firma del Documento por el Remitente (Usuario A):** Inicialmente, el remitente (Usuario A) aplica una función de resumen de mensaje (message digest function) al documento original. Esta función genera un valor de resumen (digest value). Luego, el

remitente utiliza su clave privada para cifrar el documento y adjunta una firma digital. La firma digital es un valor criptográfico que se genera a partir del documento y su clave privada y se adjunta al documento para garantizar su integridad y autenticidad (Islam et al., 2022).

- **Transmisión y Recepción del Documento**: El documento firmado se envía al receptor (Usuario B).
- **Verificación por el Receptor (Usuario B):** Al recibir el documento, el receptor (Usuario B) aplica la misma función de resumen de mensaje que se utilizó en el extremo del remitente (Usuario A) para obtener un valor de resumen. Luego, el receptor utiliza la clave pública del remitente (Usuario A) para descifrar el documento. Al hacerlo, se obtiene otro valor de resumen.
- **Comparación de Valores de Resumen**: El receptor compara los dos valores de resumen obtenidos. Si son idénticos, esto indica que el documento no ha sido alterado durante la transmisión y que la firma digital es válida. En otras palabras, la firma digital se verifica con éxito.
- **Inclusión en el Bloque de Transacciones:** Si la firma se verifica con éxito, el documento se considera auténtico y se agrega al bloque de transacciones, como se ilustra en la Figura 4.5.

Figura 4.5: Verificabilidad



(Islam et al., 2022)

4.3 Conceptos inherentes de Blockchain

4.3.1 Descentralización

La descentralización en blockchain significa que la red no depende de una autoridad central y en su lugar, la toma de decisiones y la validación de transacciones se distribuyen en una red de nodos, lo que aumenta la seguridad y la resistencia a la censura.

Blockchain fue diseñado como un sistema descentralizado, pero los mineros en proceso tienden a centralizarse. Se estima que los cinco principales grupos de minería poseen colectivamente más del 51% del poder de cómputo total de la red Bitcoin. En la figura 4.6 el caso B representa este tipo de red.

4.3.2 Distribuído

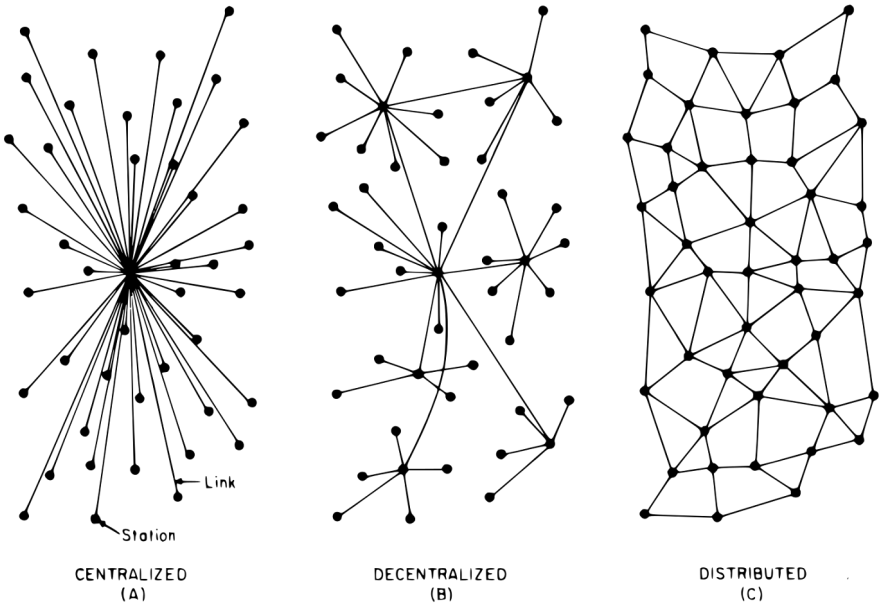
En el contexto de blockchain, una "red distribuida" se refiere a una red de nodos interconectados que trabajan juntos para mantener y validar la cadena de bloques. Cada nodo en la red tiene una copia de la cadena de bloques y participa en la validación de transacciones y en la toma de decisiones sobre el estado de la red. La distribución de la red garantiza la seguridad, la resistencia a la censura y la independencia de nodos, ya que no depende de una entidad central. En la figura 4.6 caso C se representa esta red.

4.3.3 Centralizado

En una red centralizada, todas las decisiones y el control están concentrados en una entidad central o servidor, con lo cual tiene un control total y autoridad sobre la cadena de bloques. Esta entidad central tiene autoridad sobre la red y toma decisiones para todos los usuarios y nodos. Los usuarios y nodos dependen de la entidad central para acceder a la red y llevar a cabo operaciones. Esto erosiona algunos de los principios fundamentales de

seguridad, resistencia a la censura e independencia que son inherentes a la tecnología blockchain descentralizada. En la figura 4.6 caso A se representa esta red.

Figura 4.6: Tipos de redes



4.3.4 Función hash

Una función hash emplea un algoritmo que convierte una entrada de longitud variable en un valor de hash con una longitud fija. Los valores de hash resultantes son consistentes para la misma entrada, lo que significa que siempre generan el mismo valor hash cuando se aplica la función. Además, estas funciones hash transforman los datos de entrada en una salida de tal manera que incluso las mínimas diferencias en la entrada provocan cambios significativos en el resultado.

En el ámbito informático, las funciones hash son fundamentales para la rápida recuperación de datos mediante tablas hash. En el contexto de las criptomonedas como Bitcoin, se utilizan funciones hash criptográficas, también conocidas como funciones hash seguras, para llevar a cabo tareas de consenso. Estas funciones presentan propiedades adicionales además de las características hash estándar, que incluyen:

- **Función unidireccional o resistente**: Significa que, dado un valor de hash, resulta prácticamente imposible determinar los datos de entrada originales.
- **Baja probabilidad de colisiones**: Implica que, dada una entrada, encontrar otra entrada con el mismo valor de hash debe ser un proceso computacionalmente costoso.
- **Alta resistencia a colisiones**: Se refiere a que es virtualmente imposible que dos conjuntos de datos diferentes generen el mismo valor de hash, dado un valor de hash específico.

Una función hash es como un sello digital que se aplica a un bloque de datos de transacciones. Este sello es único para cada bloque y se calcula de manera que cualquier cambio, incluso el más pequeño, en los datos del bloque generaría un sello completamente diferente. Es como si cada bloque tuviera su propia huella digital única.

Este sello digital, o valor de hash, tiene varias funciones importantes en el Blockchain. Primero, garantiza que los datos en un bloque no hayan sido alterados, lo que asegura la integridad de la información. Segundo, conecta los bloques en una cadena, ya que cada bloque contiene el valor de hash del bloque anterior, creando así una secuencia inmutable. Además, estas huellas digitales permiten realizar búsquedas y verificar datos de manera eficiente. Por último, la seguridad de estas funciones hash es crítica, ya que deben ser resistentes a intentos de alteración y garantizar la confiabilidad de la Blockchain.

4.3.5 Timestamp

Un "timestamp" en el Blockchain es una marca de tiempo que registra la fecha y hora exactas en que se añadió un bloque de transacciones a la cadena. Su importancia radica en:

- **Secuencia de Eventos**: Establece el orden cronológico de transacciones y bloques, manteniendo la integridad y el registro preciso de eventos.
- **Inmutabilidad**: Contribuye a la inmutabilidad del Blockchain al detectar cualquier intento de alterar el contenido de un bloque.

- **Búsqueda Eficiente**: Facilita la búsqueda y recuperación de datos en la cadena, permitiendo a los usuarios encontrar información específica en momentos precisos.
- **Consistencia de la Red**: En redes Blockchain públicas, las marcas de tiempo respaldan el proceso de consenso, sincronizando nodos y garantizando la seguridad de la red.

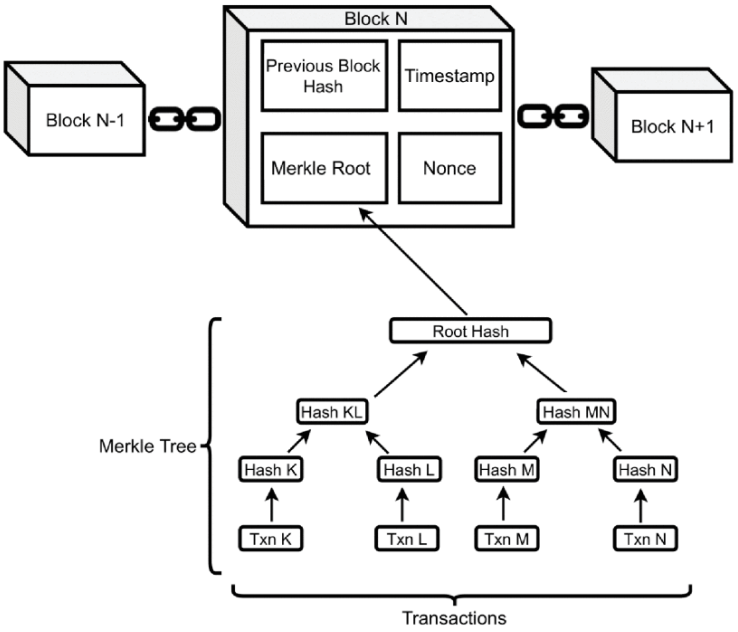
4.3.6 Merkle trees

En 1982, Ralph Merkle propuso una estructura de datos llamada "árbol Merkle" o "árbol hash". Satoshi Nakamoto adoptó la idea de los árboles Merkle para marcar con fecha y hora un bloque de transacciones en blockchain. Estos árboles son comunes en aplicaciones de intercambio de archivos para mantener un seguimiento de los bloques de archivos.

Un árbol Merkle calcula el valor hash de un bloque de transacciones de la siguiente manera: Inicialmente, se forma un árbol binario en el que cada nodo padre tiene dos hijos, y las hojas son los hashes de transacciones individuales (h0, h1, h2, ...). El hash del nodo padre se calcula como el hash de sus dos hijos, por ejemplo, h01 = H(h0), donde H es la función de hash SHA-256. Finalmente, se calcula el hash de la raíz del árbol, llamado "raíz Merkle", que se puede adjuntar al encabezado del bloque (Islam et al., 2022).

Sin embargo, un desafío importante de los árboles Merkle es la autenticación de transacciones. Si un nodo desea verificar si una transacción específica (como t3) está en un bloque, debe realizar una búsqueda que toma logarítmicamente proporcional al número de nodos (n) en el árbol. El nodo debe medir h3, h23 y el hash raíz, y luego verificar el hash raíz almacenado en el bloque. Esto significa que el nodo solo puede buscar en la rama única del árbol que contiene la transacción deseada. Las transacciones se organizan en un árbol de Merkle, crucial para verificar la coherencia e integridad de grandes conjuntos de datos transaccionales de manera eficiente y segura. (KUSHWAHA et al., 2022), en el ejemplo de la Figura 4.7.

Figura 4.7: Estructura de los nodos de una Blockchain Ethereum



(KUSHWAHA et al., 2022)

4.5 Algoritmos de consensos

El "Problema de los Generales Bizantinos" se originó en el siglo XIII cuando generales con ejércitos separados intentaron coordinar un ataque a una ciudad sitiada. La coordinación era difícil debido a la comunicación limitada a través de mensajeros y la posibilidad de traidores que podrían modificar mensajes. Este problema histórico sirve como base para los mecanismos de consenso en las cadenas de bloques, que buscan establecer acuerdos y confianza entre participantes para determinar la verdad. Los mecanismos de consenso son estrategias utilizadas en las cadenas de bloques para llegar a un acuerdo común o un estado único en la red. Existen diversas técnicas para abordar este desafío, que se describirán a continuación.

4.5.1 Algoritmo de consenso: proof of work (PoW)

En una red descentralizada, se selecciona un nodo para registrar todas las transacciones. Para evitar ataques, se utiliza el concepto de PoW. Los nodos deben realizar cálculos intensivos para demostrar que no atacarán la red. Esto se logra calculando repetidamente el hash del encabezado de un bloque, con cambios frecuentes en un valor llamado "nonce" hasta que se obtenga un hash que cumpla con un requisito específico. Una vez que un nodo alcanza este valor deseado, difunde el bloque a otros nodos, que lo validan mutuamente. Si es válido, se agrega a la cadena de bloques. Este proceso, conocido como "minería", se repite cada 10 minutos. Los nodos que contribuyen a este proceso se llaman "mineros". En términos sencillos, la minería implica adivinar un valor "nonce" que produce un hash con cierta cantidad de ceros iniciales.

4.5.2 Algoritmo de consenso: proof of stake (PoS)

Es una alternativa a PoW que asegura la red al bloquear activos criptográficos. En PoS, los "mineros" deben demostrar que poseen la criptomoneda. Los validadores en PoS apuestan sus tokens en la red al bloquearlos en la cadena de bloques para crear y aceptar bloques. Reciben compensación proporcional a la cantidad apostada, lo que incentiva la validación basada en el retorno de la inversión. Los validadores son seleccionados para crear nuevos bloques en función de su apuesta, y los bloques propuestos son verificados y añadidos a la cadena de bloques. El proceso incluye medidas para evitar la manipulación y se basa en la cantidad apostada por los validadores.

4.5.3 Algoritmo de consenso: delegated proof of stake (DPoS)

La diferencia clave entre DPoS y PoS es que DPoS es un sistema democrático representativo. En DPoS, los titulares de tokens eligen a sus representantes para crear y

validar bloques, llamados validadores. Debido a que hay menos nodos, la validación es más rápida, lo que acelera las transacciones. Además, los delegados pueden ajustar el tamaño y los intervalos de los bloques, y los usuarios pueden votar para cambiar su delegado si es necesario.

4.5.4 Algoritmo de consenso: Practical Byzantine Fault Tolerance (PBFT)

El algoritmo PBFT es una técnica de replicación diseñada para tolerar fallas tipo Bizantinas, como se vieron en el "Problema de los Generales Bizantinos", y puede manejar hasta un tercio de estas fallas. El proceso se divide en tres partes: prepreparación, preparación y compromiso. En cada ronda, se selecciona un nodo principal que es responsable de ordenar las transacciones, y un cliente envía una solicitud a este nodo principal. El nodo principal envía mensajes a todos los demás nodos, y estos nodos aceptan los mensajes si son válidos. Los mensajes a menudo incluyen firmas y números de secuencia para determinar su validez. Si un nodo acepta un mensaje de preparación, se reconoce enviando un mensaje de preparación a todos los demás nodos. Los nodos que reciben estos mensajes de preparación los aceptan si son verdaderos. Luego, un nodo se considera preparado si recibió la solicitud original del nodo principal, el mensaje de preparación y vio mensajes de preparación de otros nodos. Después de que los nodos estén preparados, envían un mensaje de compromiso al resto de los nodos. Si un nodo recibe suficientes mensajes de compromiso válidos, realiza la solicitud del cliente y envía una respuesta al cliente. El cliente espera suficientes respuestas y las considera correctas. Un nodo avanza al siguiente paso si obtiene el apoyo de dos tercios de todos los nodos. PBFT requiere que la red conozca a cada nodo y consulte a otros nodos para funcionar correctamente (Islam et al., 2022).

En la Tabla 4.1 se muestra una comparativa de estos algoritmos de consenso con criterios de uso de energía, tolerancia y la aplicación.

Tabla 4.1: Comparativa de algoritmos de consenso

Algoritmo de Consenso	Uso de Energía	Tolerancia al Poder del Adversario	Aplicación
Proof of Work (PoW)	Alto	Alto	Bitcoin
Practical Byzantine Fault Tolerance (PBFT)	Bajo	Alto	Hyperledger Fabric
Proof of Stake (PoS)	Bajo	Alto	Ethereum
Delegated Proof of Stake (DPoS)	Bajo	Alto	EOSIO

- **Uso de Energía**: Un algoritmo se considera que tiene un “alto” uso de energía si consume una gran cantidad de recursos para realizar sus operaciones. Por otro lado, un “bajo” uso de energía significa que el algoritmo es eficiente y minimiza el uso de recursos.
- **Tolerancia al Poder del Adversario**: Un algoritmo tiene una “alta” tolerancia al poder del adversario si puede resistir ataques o interferencias significativas de un adversario sin comprometer su funcionalidad. Una “baja” tolerancia significa que el algoritmo puede ser susceptible a interferencias o ataques.

4.6 Tipos de Blockchain

4.6.1 Redes Privadas

Una red privada de blockchain es como una empresa con un grupo selecto de empleados. El acceso a esta red está restringido y controlado, generalmente por una

organización o entidad central. Solo las partes autorizadas pueden unirse a la red y participar en las transacciones. Esta red se asemeja a una empresa que maneja sus operaciones internas con un alto nivel de privacidad y confidencialidad. La seguridad en una red privada se basa en la confianza mutua entre los participantes, ya que estos se conocen y verifican entre sí.

4.6.2 Redes Públicas

Una red pública de blockchain como una plaza pública digital. Es un espacio abierto en línea al que cualquiera puede acceder sin restricciones. Cualquier individuo o entidad puede unirse a esta red y participar en la verificación y registro de transacciones. Al igual que en una plaza pública, todas las transacciones y registros son visibles para todos. La seguridad en esta red se basa en el consenso de una comunidad global de nodos que validan las transacciones, lo que garantiza la integridad de la red.

4.6.3 Redes Consorcio

Una red de consorcio de blockchain es como un proyecto conjunto entre varias organizaciones de confianza. Estas organizaciones comparten la responsabilidad y el control de la red, actuando como socios en un esfuerzo colaborativo. Aunque el acceso está restringido y sujeto a aprobación, no depende de una entidad central. Es como un grupo de organizaciones que se unen para alcanzar un objetivo común. La seguridad en esta red se basa en la colaboración entre organizaciones de confianza que trabajan juntas para validar y registrar transacciones.

4.7 Resumen

Se exploró en profundidad los conceptos fundamentales relacionados con la tecnología blockchain. Se definió lo que es el blockchain y sus conceptos inherentes, incluyendo la

descentralización, la distribución, y la centralización. También se analizó la importancia de las funciones hash, los timestamps y las estructuras de Merkle trees en el contexto del blockchain.

Además, se presentaron los algoritmos de consenso que son esenciales para el funcionamiento de una red blockchain, incluyendo PoW, PoS, DPoS y PBFT. Estos algoritmos desempeñan un papel crucial en la validación de transacciones y el mantenimiento de la integridad de la red.

Finalmente, se abordó los diferentes tipos de blockchains, que abarcan desde las redes privadas, diseñadas para aplicaciones internas y la colaboración empresarial, hasta las redes públicas, que son abiertas a cualquiera. También se consideraron las redes de consorcio, que son proyectos colaborativos entre organizaciones de confianza.

Este conocimiento sobre los fundamentos de blockchain sienta las bases para comprender cómo esta tecnología está transformando diversas industrias y está siendo utilizada para innovar en una amplia gama de aplicaciones. En los siguientes capítulos, se profundizará su implementación y aplicaciones en el mundo real.

Capítulo 5 Smart Contracts

5.1 Introducción

Uno de los avances más destacados en este panorama es la introducción de los "smart contracts" o contratos inteligentes (SC). Estos contratos, basados en la tecnología blockchain, representan un hito revolucionario al transformar la ejecución de acuerdos y transacciones de manera automatizada, transparente y segura. Este capítulo se sumerge en el mundo de los smart contracts, explorando sus fundamentos esenciales y su aplicación práctica. Comenzaremos desentrañando la diferencia clave entre los contratos tradicionales y los smart

contracts, destacando cómo esta innovación ha eliminado barreras y ha redefinido la confianza en las transacciones digitales.

A lo largo de estas páginas, entenderemos cómo la turing completeness y la programación en lenguajes específicos, como Solidity, permiten la expresividad y versatilidad de estos contratos autónomos. La plataforma Ethereum, pionera en la implementación de smart contracts, será analizada. Además, se abordarán las funciones prácticas de los smart contracts, desde la gestión de tokens y acuerdos de votación hasta sistemas de entidad descentralizada. Ejemplos concretos ilustran cómo estos contratos han revolucionado la forma en que concebimos y ejecutamos acuerdos digitales. Sin embargo, no se ignorarán los desafíos y riesgos asociados con esta tecnología. La seguridad en smart contracts, vital dada la irreversibilidad de las transacciones en blockchain, será analizada en profundidad. Se destacarán los problemas comunes y las mejores prácticas para mitigarlos.

Finalmente, este capítulo explorará el panorama actual de la adopción de smart contracts, identificando las industrias que lideran el cambio y señalando los desafíos que aún deben superarse. Desde finanzas hasta logística, estos contratos inteligentes están impactando múltiples sectores y redefiniendo la manera en que concebimos y ejecutamos acuerdos digitales.

5.2 Concepto de Smart Contract

Los contratos inteligentes son un avance importante en la tecnología blockchain. Propuestos en la década de 1990 como un protocolo digital para cumplir acuerdos, son contenedores de código que replican términos contractuales en el ámbito digital. Son acuerdos vinculantes entre partes, reemplazando intermediarios con ejecución automática en una cadena de bloques descentralizada. Permiten transacciones entre partes no confiables sin intermediarios. Comparados con contratos convencionales, reducen riesgos, costos

administrativos y mejoran eficiencia corporativa al ser alojados en una cadena de bloques. Proyectan mejorar el mecanismo de transacción actual en varias industrias (Taherdoost, 2023).

5.2.1 Contratos Tradicionales vs. Smart Contracts

La evolución digital ha transformado radicalmente la manera en que celebramos acuerdos y gestionamos transacciones. Los contratos tradicionales, pilares del intercambio comercial y legal, han coexistido con una nueva forma de contratación que está redefiniendo las reglas del juego: los smart contracts o contratos inteligentes.

En el pasado, los contratos tradicionales, aunque fundamentales, presentaban desafíos inherentes. Estos documentos legales requerían la intervención humana para su ejecución y a menudo dependían de intermediarios para garantizar el cumplimiento de los términos acordados. La burocracia, los plazos prolongados y la necesidad de confiar en terceros eran obstáculos comunes en este paradigma contractual.

En contraste, los smart contracts introducen una era de automatización, transparencia y eficiencia. Estos programas informáticos, ejecutados de manera autónoma en una blockchain, eliminan la necesidad de intermediarios y garantizan la ejecución automática de los términos contractuales cuando se cumplen condiciones predefinidas. La intervención humana se reduce al mínimo, y la confianza se establece a través de la inmutabilidad y transparencia inherentes de la tecnología blockchain.

La implementación de contratos inteligentes y tradicionales refleja dos enfoques distintos en la gestión de transacciones y acuerdos. Los contratos inteligentes, al ser autoejecutables y basados en blockchain, ofrecen seguridad e inmutabilidad, reduciendo riesgos de fraude y costos de transacción. Sin embargo, su adopción generalizada se ve limitada por desafíos de reconocimiento legal y la posibilidad de errores en la programación. Por otro lado, los contratos tradicionales, respaldados por la protección legal y flexibilidad,

proporcionan registros permanentes y son accesibles para un público más amplio. A pesar de su vulnerabilidad a modificaciones y procesos manuales, la confianza en su ejecución sigue siendo esencial. La elección entre ambos dependerá de consideraciones específicas de cada situación, equilibrando las ventajas tecnológicas con las garantías legales y la madurez tecnológica en la jurisdicción correspondiente. A continuación, en la Tabla 5.1 se muestran las diferencias esenciales de los contratos tradicionales y los contratos inteligentes.

Tabla 5.1: Diferencias entre los Smart Contracts y los Contratos Tradicionales

Factor	Contratos inteligentes	Contratos Tradicionales
Legalidad	Legalmente vinculantes solo si cumplen con las leyes aplicables.	Un acuerdo legalmente vinculante que se puede hacer cumplir en los tribunales.
Ejecución	Se ejecuta y se aplica automáticamente cuando se cumplen los términos del contrato.	Ejecutado y ejecutado a través del sistema judicial.
Validez	Puede ser validado por cualquier persona en la red blockchain.	Válido solo si ambas partes están de acuerdo y firman el contrato.
Modificación	Difícil de modificar una vez desplegado.	Fácilmente modificable y enmendado con el consentimiento mutuo de ambas partes.
Mantenimiento de registros	Los registros se almacenan en una cadena de bloques y se pueden ver públicamente.	Registros almacenados en papel o digitalmente, no disponibles públicamente.
Costar	Rentable gracias a la eliminación de intermediarios externos.	Requieren el uso de intermediarios externos, lo que los hace más costosos.
Seguridad	Altamente seguro ya que se almacenan en una cadena de bloques.	Menos seguros ya que se almacenan en papel o digitalmente.
Transparencia	Altamente transparente, ya que todas las partes involucradas pueden ver la cadena de bloques.	No es transparente, ya que solo las partes involucradas pueden ver el acuerdo.
Trazabilidad	Todas las actividades son rastreables en la cadena de bloques.	El rastreo de las actividades es difícil, ya que el contrato se almacena en papel o digitalmente.
Exactitud	Automatizado y preciso gracias al uso de código.	Manual y propenso a errores debido a la intervención humana.

(Jain, 2023)

5.2.2 Turing Completeness

En el núcleo conceptual de los smart contracts se encuentra la noción de Turing Completeness, un principio que delinea la capacidad de un lenguaje de programación, y por ende de un smart contract, para realizar cualquier cálculo algorítmico. Esta idea, establecida por Alan Turing en la teoría de la computación, postula que cualquier tarea computable puede ser llevada a cabo por una máquina de Turing, un principio que se traslada con impacto al mundo de los contratos inteligentes. Esta característica es de vital importancia, ya que proporciona a los smart contracts una versatilidad excepcional. La capacidad Turing Completeness permite la implementación de una amplia gama de operaciones, desde simples cálculos matemáticos hasta la ejecución de algoritmos complejos. En el contexto de las plataformas blockchain, lenguajes como Solidity se consideran Turing Completeness, permitiendo la creación de smart contracts que operan de manera similar a cualquier programa informático.

La flexibilidad inherente a la Turing Completeness es un habilitador clave para el desarrollo de aplicaciones descentralizadas avanzadas. Desde contratos financieros sofisticados hasta sistemas de votación autónomos, la versatilidad de los smart contracts transforma la manera en que concebimos y desarrollamos soluciones en el entorno blockchain.

5.2.3 dApps

Las dApps, o aplicaciones descentralizadas, son programas que se ejecutan en redes descentralizadas, principalmente basadas en tecnología blockchain. Operan sin un control central, utilizan contratos inteligentes para automatizar procesos, garantizan transparencia mediante registros en la cadena de bloques, y suelen hacer uso de tokens nativos de la red. Destacan por su resistencia a la censura y su capacidad para funcionar sin intermediarios.

5.3 Ethereum

Desde su concepción en 2013 y lanzamiento en 2015, Ethereum ha liderado la evolución de los contratos inteligentes, para más interés sobre su aparición se puede encontrar en el Anexo I. Su Definición, una plataforma blockchain que introdujo los smart contracts de forma nativa. Más que una criptomoneda, Ethereum se propuso como una plataforma para ejecutar estos contratos y fomentar el desarrollo dApps. Con Solidity, un lenguaje Turing Completeness, Ethereum permitió la creación de smart contracts con funcionalidades avanzadas, abriendo la puerta a una versatilidad sin precedentes. Esta visión no solo buscaba la ejecución de contratos, sino también ser el pilar para el desarrollo de dApps en áreas como en las finanzas descentralizadas (DeFi), juegos y redes sociales.

Afrontando desafíos como la escalabilidad y las tarifas de transacción, Ethereum 2.0 con la PoS apunta a solucionar estos problemas. El impacto de Ethereum se extiende más allá de su red, inspirando proyectos y estableciendo estándares como los tokens.

En su continuo desarrollo, Ethereum sigue siendo esencial. La actualización Ethereum 2.0 promete mejorar la escalabilidad y sostenibilidad, influenciando no solo la red Ethereum sino también el futuro de los contratos inteligentes y las dApps en la blockchain. Ethereum ha dejado una huella indeleble, siendo un impulsor clave en la adopción de los contratos inteligentes.

5.3.1 Máquina Virtual de Ethereum

La Ethereum Virtual Machine (EVM) es un componente central e innovador en la plataforma blockchain de Ethereum. Funciona como una máquina virtual descentralizada que ejecuta y valida contratos inteligentes, representando un pilar fundamental para la creación de dApps y la ejecución de contratos autoejecutables en la red Ethereum (*Ethereum Homestead, n.d.*).

La EVM opera como una capa de ejecución universal en todos los nodos de la red Ethereum, garantizando la coherencia y consistencia en la ejecución de código a lo largo de la blockchain. Su diseño Turing completo significa que es capaz de ejecutar cualquier algoritmo o programa expresado de manera algorítmica, brindando a los desarrolladores un entorno potente y flexible para la implementación de contratos inteligentes.

Cuando un contrato inteligente se despliega en la red Ethereum, la EVM interpreta y ejecuta su código de manera determinista, asegurando que todos los nodos lleguen al mismo resultado y manteniendo un consenso descentralizado sobre el estado del contrato. Esta característica es esencial para la confiabilidad y la seguridad de la ejecución de contratos en un entorno distribuido.

5.3.2 Ether

El Ether (ETH) representa la criptomoneda principal vinculada a la plataforma Ethereum, exhibiendo similitudes con el Bitcoin pero diseñado para operar en entornos digitales. A continuación, se detallan aspectos significativos del Ether (*Ethereum Homestead*, n.d.):

- Gestión Autónoma: El Ether concede la capacidad de actuar como entidad financiera propia, permitiendo el control total de los fondos a través de una cartera como evidencia de propiedad, prescindiendo de terceros intermediarios.
- Resguardo Criptográfico: A pesar de la novedad del dinero digital, el Ether se respalda en técnicas criptográficas consolidadas, garantizando la seguridad de la cartera, los ETH y cada transacción.
- Transferencias P2P: Las transacciones con ETH se realizan de igual a igual, prescindiendo de intermediarios como bancos. Este método se asemeja a la entrega de dinero en efectivo en persona, pero se ejecuta de manera segura y global en cualquier momento y lugar.

- Descentralización: ETH se caracteriza por ser una moneda global y descentralizada, excluyendo la influencia de empresas o bancos. Esto impide la generación arbitraria de más ETH o modificaciones centralizadas en los términos de uso.
- Acceso Universal: La aceptación de ETH únicamente requiere conexión a Internet y una cartera, prescindiendo de la necesidad de una cuenta bancaria para recibir pagos.
- Divisibilidad Flexible: Ether es divisible hasta 18 decimales, permitiendo la adquisición de fracciones, incluso mínimas, como 0.000000000000000001 ETH.

Cabe destacar que Ethereum constituye la infraestructura implícita, mientras que ETH se erige como su activo principal. Al llevar a cabo transacciones en Ethereum, se abona una pequeña tarifa en ETH, actuando como incentivo para que los productores de bloques procesen y verifiquen las transacciones en la red.

Ethereum utiliza un sistema de unidades específico para el ether, donde cada denominación posee un nombre único. El Wei representa la unidad de medida más diminuta, en la Tabla 5.2 se encuentran las equivalencias de cada unidad de medida.

Tabla 5.2: Unidades de medida del Ether

Unidad	**Equivalencia**
Wei	1 Wei
Lovelace	1,000 Wei
Babbage	1,000 Lovelace
Shannon	1,000 Babbage
Szabo	1,000 Shannon

(Ulrich)

Cada unidad superior es 1,000 veces la unidad inferior. Y recuerda, cada unidad de Ether puede dividirse hasta en 10e-18 unidades. Esto proporciona una gran flexibilidad para las transacciones y contratos inteligentes en la red Ethereum.

5.3.3 Concepto de Minado

En el ecosistema de las criptomonedas y las blockchains, los mineros desempeñan un papel vital en la validación y seguridad de la red. Su función principal consiste en confirmar transacciones y agregar nuevos bloques a la cadena de bloques, un proceso conocido como minería (*Ethereum Homestead*, n.d.).

En sistemas basados en PoW, como el utilizado por Bitcoin, los mineros se dedican a resolver complejos problemas matemáticos que requieren una considerable potencia de cómputo. El primer minero en resolver con éxito el problema tiene el derecho de añadir el nuevo bloque a la cadena y es recompensado con nuevas unidades de la criptomoneda nativa, así como las tarifas asociadas con las transacciones contenidas en el bloque (*Ethereum Homestead*, n.d.).

La recompensa sirve como un incentivo crucial para que los mineros participen en la red y ofrezcan la potencia de procesamiento necesaria. Además de validar transacciones, la participación distribuida de mineros contribuye a la seguridad de la red. Modificar la blockchain requeriría que un atacante controlara más del 50% del poder de cómputo total, un escenario poco probable y costoso conocido como ataque del 51%. Un componente esencial del proceso de minería es el "nonce" (número arbitrario utilizado solo una vez). El nonce es un valor que los mineros ajustan repetidamente en su intento de resolver el problema criptográfico. Este proceso de ajuste del nonce es lo que otorga la propiedad de la Prueba de Trabajo al sistema, ya que los mineros deben demostrar que han realizado un trabajo computacional significativo (*Ethereum Homestead*, n.d.).

5.3.4 Gas

El término "gas" se refiere a la unidad de medida utilizada para cuantificar la cantidad de recursos computacionales necesarios para llevar a cabo una operación o ejecutar un contrato inteligente en la red. En otras palabras, el gas representa el costo computacional de realizar acciones dentro de la plataforma Ethereum (*Ethereum Homestead*, n.d.).

Cuando un usuario desea realizar una transacción o ejecutar un contrato inteligente en Ethereum, debe pagar una cierta cantidad de gas. Este gas actúa como una tarifa que compensa a los nodos de la red por el trabajo realizado al procesar y validar la operación. Cada operación en Ethereum consume una cantidad específica de gas, y la tarifa en ether (ETH) asociada con la operación se calcula multiplicando la cantidad de gas por el precio del gas, expresado en términos de ether por unidad de gas (*Ethereum Homestead*, n.d.).

El sistema de gas en Ethereum tiene varios propósitos clave (*Ethereum Homestead*, n.d.):

- **Incentivos para los Mineros/Nodos**: Los mineros y nodos de la red son recompensados con las tarifas de gas por su participación en la ejecución y validación de operaciones. Esto proporciona un incentivo económico para mantener la red y garantizar su seguridad.
- **Prevención de Ataques**: Al asociar costos con operaciones específicas, el sistema de gas evita abusos y ataques de denegación de servicio. Los usuarios deben pagar por el uso de recursos computacionales, lo que desincentiva comportamientos maliciosos.
- **Optimización de Recursos**: Los desarrolladores pueden optimizar sus contratos y transacciones para reducir el consumo de gas y, por lo tanto, minimizar las tarifas asociadas. Esto fomenta la eficiencia y la mejora continua de las prácticas de desarrollo en la red Ethereum.

5.3.5 Transacciones

Las transacciones se identifican a través de un hash y quedan registradas en un bloque cuando el estado es exitoso. Esto incluye el momento de generación, las cuentas involucradas, la dirección del contrato inteligente, el costo de la transacción y el precio del Gas empleado durante la transacción. El costo de la transacción está intrínsecamente relacionado con el precio del Gas y la cantidad de Gas consumida. El Gas representa la medida del poder computacional requerido para llevar a cabo una transacción en la cadena de bloques. A medida que aumenta la complejidad de una operación y se demanda un mayor poder computacional para su procesamiento en la blockchain, el costo en términos de Gas será proporcionalmente mayor. Cada operación en un contrato inteligente implica el consumo de una cantidad específica de Gas, y los usuarios deben abonar este Gas utilizando la criptomoneda nativa de la cadena de bloques (por ejemplo, Ether en Ethereum). Este costo varía en función de la demanda de la red y la complejidad de la operación, lo que hace que el costo de transacción sea variable. Los usuarios establecen un límite de Gas (Gas Limit) al enviar una transacción, si están dispuestos a pagar más, su transacción tiene una mayor probabilidad de procesarse con mayor rapidez.

5.3.6 Funciones y Ejemplos

Las funciones representan las operaciones y lógicas programadas que los contratos pueden ejecutar. Estas funciones permiten la interacción entre los participantes y la automatización de acuerdos. Se exploran algunas funciones comunes y proporcionamos ejemplos para su aplicación práctica.

Funciones Comunes

- **Transferencia de Fondos:** Una función esencial permite la transferencia de fondos entre participantes. Por ejemplo, una función "transferirFondos" puede facilitar pagos automáticos cuando se cumplen ciertas condiciones.
- **Registro de Datos:** Funciones que registran información en la cadena de bloques. Por ejemplo, una función "registrarEvento" podría almacenar detalles sobre una transacción o evento específico.
- **Condicionales y Lógica de Negocios:** Las funciones pueden contener lógica condicional. Por ejemplo, una función "aprobarPréstamo" podría ejecutarse solo si ciertos criterios, como la verificación crediticia, son favorables.

Ejemplos Prácticos

- **Contratos de Apuestas Descentralizadas:** Un smart contract puede facilitar apuestas sin la necesidad de intermediarios. Una función "realizarApuesta" podría manejar el proceso de aceptación de apuestas y distribuir automáticamente los fondos al ganador.
- **Gestión de Suministros en la Cadena de Suministro:** Un smart contract puede rastrear automáticamente el movimiento de productos. Una función "actualizarUbicación" podría registrarse cada vez que un producto se mueve a través de la cadena de suministro.
- **Tokens No Fungibles (NFTs):** Los smart contracts se utilizan para crear y gestionar tokens no fungibles. Una función "transferirNFT" podría permitir la transferencia de un token único de un usuario a otro.

5.5 Solidity: Creando Contratos Inteligentes

Solidity es un lenguaje de programación de alto nivel orientado a objetos. Se utiliza principalmente en aplicaciones alojadas en la red de Ethereum. Su uso más destacado es en la construcción de contratos inteligentes para aplicaciones basadas en tecnología blockchain.

Estos contratos inteligentes pueden albergar archivos PDF, en los que se determinen ciertas condiciones entre las partes firmantes o en los que se tengan en cuenta otros aspectos relevantes. Como lenguaje de alto nivel, Solidity genera software que es intuitivamente comprensible sin hacer referencias directas al hardware. Esto significa que los desarrolladores pueden centrarse en la lógica de la aplicación sin preocuparse por los detalles de bajo nivel. Además, Solidity es un lenguaje orientado a objetos, lo que significa que puede manejar en su estructura de código grandes referencias a datos. Estos datos pueden ser cálculos, bases de datos u objetos predefinidos (como archivos Word o PDF), entre otros (*Solidity Programming Language*, n.d.).

5.6 Oráculos: Conectando el Mundo Real con la Blockchain

Los smart contracts son poderosos para ejecutar lógica programable en la blockchain pero tienen una limitación fundamental, la incapacidad de acceder directamente a información fuera de la cadena. Acá es donde entran en juego los oráculos, actuando como intermediarios para llevar datos del mundo real a los contratos inteligentes y viceversa.

Los oráculos son entidades externas que proporcionan datos del mundo real a los contratos inteligentes. Estos datos pueden incluir información sobre precios de activos, resultados de eventos deportivos, datos meteorológicos, entre otros. Al integrar oráculos, los smart contracts pueden tomar decisiones basadas en información actualizada y relevante.

5.6.1 Funcionamiento de los Oráculos

1. Solicitud de Información: Un usuario o un smart contract solicita información a un oráculo, especificando el tipo de datos requeridos.

2. Recopilación de Datos: El oráculo recopila la información del mundo real a través de fuentes externas, como APIs, sensores o incluso datos ingresados manualmente.
3. Envío a la Blockchain: Una vez recopilada la información, el oráculo la transmite a la blockchain, donde el smart contract puede acceder a ella.
4. Ejecución del Smart Contract: Con los datos del oráculo, el smart contract puede ejecutar su lógica programada y tomar decisiones basadas en información actualizada.

Ejemplos de Uso:

- **Contratos Financieros:** Los oráculos son fundamentales en contratos financieros que dependen de datos del mercado, como tasas de cambio o precios de acciones.
- **Seguros Basados en Eventos:** Los oráculos pueden verificar eventos específicos (como vuelos retrasados) y activar automáticamente pagos de seguros si se cumplen las condiciones.
- **Juegos Descentralizados:** En juegos basados en blockchain, los oráculos pueden proporcionar resultados de eventos del mundo real que afectan el desarrollo del juego.

Existen varios tipos de Oráculos, incluyendo:

- **Oráculos Centralizados:** Operados por entidades centralizadas. Sin embargo, pueden generar problemas de confianza y centralización.
- **Oráculos Descentralizados:** Utilizan múltiples fuentes de datos descentralizadas para brindar información a los contratos. Esto evita depender de una única fuente.
- **Oráculos Basados en Protocolos:** Son específicos de ciertas cadenas de bloques y emplean protocolos para gestionar los datos. Pueden involucrar votaciones o incentivos para asegurar la precisión de la información.

La confiabilidad de los oráculos depende de la calidad de las fuentes de datos externas. Se deben tomar medidas para garantizar la integridad y autenticidad de la información. Los

oráculos pueden ser vulnerables a ataques o manipulaciones, lo que podría afectar la ejecución de los contratos inteligentes. Se buscan soluciones para mitigar estos riesgos.

5.7 Seguridad en Smart Contracts: Protegiendo la Ejecución Descentralizada

La seguridad en smart contracts es una consideración crítica dada su ejecución automática y descentralizada. Errores de codificación, vulnerabilidades y ataques pueden tener consecuencias significativas. Exploramos los desafíos de seguridad y las mejores prácticas para garantizar la integridad de los contratos inteligentes.

Desafíos de Seguridad:

1. Errores de Codificación: Pequeños errores en el código pueden tener consecuencias graves. La falta de una corrección exhaustiva puede resultar en vulnerabilidades explotables.
2. Ataques de Reentrada: Los ataques de reentrada ocurren cuando un contrato llama a otro antes de completar su ejecución, permitiendo que el segundo contrato vuelva a entrar en el primero. Esto puede llevar a pérdidas financieras.
3. Vulnerabilidades en el Gas: El uso inadecuado del gas puede dar lugar a ataques de DoS, donde un atacante consume todo el gas disponible, bloqueando la ejecución del contrato.

Mejores Prácticas de Seguridad:

1. Pruebas Rigurosas: Realizar pruebas exhaustivas, incluyendo pruebas de penetración, para identificar y corregir posibles vulnerabilidades.
2. Revisión por Pares: Obtener revisiones de código por parte de otros desarrolladores experimentados para detectar posibles errores.

3. Uso de Bibliotecas Seguras: Utilizar bibliotecas y contratos estándar bien establecidos para evitar reinventar la rueda y reducir riesgos.
4. Control de Acceso: Implementar controles de acceso adecuados para limitar quién puede interactuar con el contrato y ejecutar ciertas funciones.
5. Manejo de Gas: Gestionar cuidadosamente el consumo de gas para evitar ataques de DoS y garantizar la eficiencia en la ejecución.

5.7.1 Auditorías de Seguridad

Realizar auditorías de seguridad por parte de expertos en contratos inteligentes es una práctica esencial. Estas auditorías pueden identificar posibles vulnerabilidades y proporcionar recomendaciones para mejorar la seguridad del contrato.

5.8 Adopción y Desafíos de los Smart Contracts

La adopción de smart contracts ha experimentado un crecimiento significativo, transformando la forma en que se llevan a cabo las transacciones y se ejecutan los acuerdos. Sin embargo, este avance no está exento de desafíos. Analizamos el estado actual de la adopción y los obstáculos que enfrenta esta tecnología innovadora.

Estado Actual de la Adopción:

- **DeFi**: Los smart contracts han impulsado el auge de DeFi, permitiendo servicios financieros descentralizados como préstamos, intercambio y staking.
- **NFTs y Entretenimiento**: La NFTs ha revolucionado la industria del entretenimiento, permitiendo la propiedad digital única y la monetización de contenido creativo.
- **Contratos Legales y Gubernamentales**: La exploración de smart contracts en contratos legales y gubernamentales ha ganado impulso, con la posibilidad de automatizar procesos y garantizar la transparencia.

Desafíos Actuales:

- **Escalabilidad:** El aumento en la demanda de transacciones en blockchain ha destacado los desafíos de escalabilidad, con congestiones y tarifas de gas elevadas en redes populares.
- **Interoperabilidad:** La falta de interoperabilidad entre diferentes blockchains y plataformas puede obstaculizar la fluidez de los smart contracts en entornos diversos.
- **Seguridad:** A pesar de las mejores prácticas, la seguridad sigue siendo un desafío. Los errores de codificación y las vulnerabilidades pueden resultar en pérdidas financieras significativas.

Perspectivas Futuras y Superación de Desafíos:

- **Evolución Tecnológica:** Mejoras en la tecnología blockchain, como Ethereum 2.0, buscan abordar desafíos de escalabilidad y costos.
- **Estándares y Regulación:** El establecimiento de estándares y regulaciones claras puede fomentar la confianza y la adopción generalizada al tiempo que mitiga riesgos legales y de seguridad.
- **Educación y Conciencia**: La educación continua sobre smart contracts y blockchain es esencial para superar barreras de adopción, tanto entre desarrolladores como entre usuarios finales.

5.9 Resumen

El análisis exhaustivo de los smart contracts revela su capacidad para transformar fundamentalmente la forma en que concebimos y ejecutamos acuerdos. Contrapuestos a los contratos tradicionales, los smart contracts destacan por su ejecución automática, eliminando intermediarios y proporcionando transparencia e inmutabilidad. La turing completeness, esencial para la expresividad de estos contratos, se encuentra en el núcleo de su versatilidad.

Ethereum, como pionera en la integración nativa de smart contracts, ha liderado el camino en su desarrollo y adopción. A través de funciones y ejemplos diversos, desde la gestión de tokens hasta sistemas de votación descentralizados, los smart contracts han encontrado aplicaciones en una amplia gama de industrias.

La introducción de oráculos ha facilitado la conexión entre los smart contracts y datos del mundo real, ampliando su alcance. Sin embargo, la seguridad emerge como una preocupación primordial, dado que la irreversibilidad de las transacciones en blockchain resalta la importancia de abordar problemas comunes. A pesar de enfrentar desafíos como la escalabilidad y la interoperabilidad, los smart contracts han experimentado una creciente adopción en industrias clave como finanzas, seguros y logística. La superación de estos desafíos, junto con el establecimiento de estándares claros y esfuerzos educativos continuos, allanará el camino para su adopción generalizada, marcando una revolución digital en la ejecución de acuerdos.

Capítulo 6 IoV y Smart Contracts

6.1 Introducción

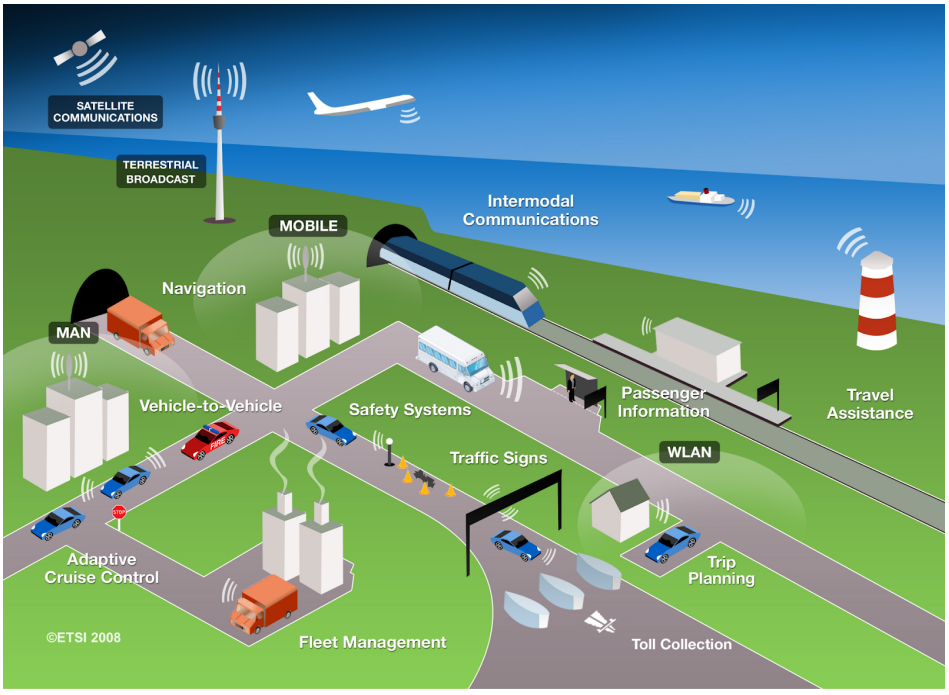
La convergencia de la tecnología Blockchain y la IoT ha dado lugar a un innovador panorama en la gestión de transporte inteligente. Este capítulo explora cómo los Smart Contracts, una aplicación fundamental de la tecnología blockchain, se integran en entornos vehiculares basados en plataformas IoT para ofrecer soluciones avanzadas y eficientes. Respaldadas por Smart Contracts en entornos vehiculares, no solo aumentan la eficiencia y la seguridad, sino que también allanan el camino para el desarrollo de sistemas de transporte más inteligentes y sostenibles. Cada sección profundiza en los casos de uso específicos y destaca el impacto positivo que estas soluciones pueden tener en la industria del transporte.

En el ámbito de la logística, por ejemplo, estas tecnologías pueden mejorar significativamente la trazabilidad y la gestión de la cadena de suministro. Al utilizar la capacidad de registro inmutable de la blockchain, se pueden rastrear productos y activos en tiempo real a lo largo de toda la cadena de suministro. Esto no solo proporciona transparencia, sino que también ayuda a reducir fraudes y pérdidas.

En el sector de la energía, la integración de blockchain e IoT puede facilitar un mercado de energía más eficiente y sostenible. Los dispositivos IoT, como medidores inteligentes, pueden recopilar datos en tiempo real sobre el consumo de energía. Estos datos se pueden almacenar de manera segura en una cadena de bloques, lo que permite una gestión más eficiente y transparente de la oferta y la demanda de energía. Además, los usuarios pueden participar activamente en la compra y venta de energía a través de contratos inteligentes.

En agricultura, la combinación de estas tecnologías puede llevar a la agricultura de precisión. Los sensores IoT en el campo pueden recopilar datos sobre condiciones climáticas, niveles de humedad del suelo y otros parámetros importantes. Estos datos se pueden almacenar de manera segura en una cadena de bloques, proporcionando a los agricultores y otros participantes en la cadena de suministro acceso a información precisa y verificable. Esto puede mejorar la toma de decisiones, aumentar la eficiencia y reducir el desperdicio.

6.2 Sistema de Gestión de Transporte Inteligente



(*The Future of Intelligent Transport Systems (ITS) | Engaged IT for the CIO*, 2011)

Los sistemas de gestión de transporte inteligente (ITS) están revolucionando la forma en que se administra y optimiza el transporte. Estos sistemas utilizan tecnologías avanzadas para mejorar la eficiencia y seguridad de las operaciones de transporte. Un enfoque prometedor en proyectos similares es la integración del blockchain y los contratos inteligentes en las ITS. Estas tecnologías ofrecen transparencia, seguridad y eficiencia en transacciones y operaciones de transporte.

Una cadena de suministro se vuelve "inteligente" cuando sus componentes pueden intercambiar datos y tomar decisiones basadas en ellos. Un contrato inteligente incluye valor, dirección, funciones y estado. En este contexto, la combinación de blockchain y contratos inteligentes en una cadena de suministro permite la verificación y validación en un entorno seguro. Debido a la globalización y la complejidad moderna, la gestión de cadenas de suministro enfrenta desafíos como decisiones ineficientes, múltiples terceros y falta de confianza. La adopción de blockchain e IoT puede resolver estos problemas. Una red

blockchain inmutable y segura se combina con dispositivos de IoT como sensores y etiquetas RFID para rastrear y administrar activos de manera encriptada. Esto promueve decisiones informadas, reduce tiempos de transporte, mejora el seguimiento en tiempo real y beneficia a las partes involucradas y a los consumidores. La colaboración en tiempo real y la infraestructura inteligente también reducen retrasos en pedidos y mejoran el control de inventario.

6.3 Aplicaciones de Smart Contracts en Entornos Vehiculares en Plataformas IoT

En la intersección entre los Smart Contracts y las plataformas IoT, emerge un horizonte de posibilidades en la optimización de procesos empresariales. Los Smart Contracts, como representantes programables de acuerdos contractuales, se fusionan con los datos y la automatización ofrecida por la IoT, generando un paradigma donde la eficiencia, la precisión y la confiabilidad se convierten en la norma. En este contexto, exploramos diversos casos que ejemplifican las ventajas concretas de esta convergencia tecnológica, demostrando cómo los Smart Contracts empoderan a las organizaciones para automatizar, agilizar y asegurar una variedad de procesos interorganizacionales.

6.3.1 Monitorización de Niveles de Combustible

Los Smart Contracts se despliegan para automatizar y supervisar los niveles de combustible en vehículos. Esta aplicación no solo mejora la eficiencia operativa, sino que también previene pérdidas y optimiza el consumo de combustible.

Suponiendo que la Empresa A y la Empresa B forman parte de una misma red blockchain. La Empresa A se dedica al transporte de activos y cuenta con un dispositivo IoT para la detección de nivel de combustible, mientras que la Empresa B es proveedora de

combustible. Utilizando la plataforma IoT y agentes configurados en el sensor de nivel de combustible, la Empresa B puede cargar el tanque del vehículo de la Empresa A. Una vez que se valide el agente, se puede generar un Smart Contract que confirme la carga completa del tanque mediante el sensor, activando una transacción de criptomoneda desde la Empresa A hacia la Empresa B como pago del combustible. Esta operación automatizada gestiona el proceso de pago y recepción del producto, en este caso, el combustible.

6.3.2 Mantenimiento de Dispositivos ("Things")

En el contexto vehicular, los Smart Contracts se utilizan para gestionar y agendar el mantenimiento de dispositivos incorporados. Esto garantiza un rendimiento óptimo, prolonga la vida útil de los equipos y reduce los costos de reparación inesperados.

En este caso, para asegurar el óptimo estado de estos dispositivos, los sensores IoT pueden recopilar datos sobre su rendimiento y salud. Los Smart Contracts podrían programar automáticamente citas de mantenimiento basadas en el análisis de estos datos. Al alcanzar umbrales específicos, el contrato podría generar solicitudes de servicio y programar mantenimientos preventivos.

6.3.3 Geoposicionamiento de Envíos

Los Smart Contracts facilitan el seguimiento y geoposicionamiento de envíos en tiempo real, mejorando la visibilidad logística y permitiendo una gestión más eficaz de las cadenas de suministro.

Siguiendo las premisas del primer caso, la Empresa A podría utilizar sensores IoT en los paquetes o contenedores para rastrear su ubicación. Los Smart Contracts podrían activarse al llegar un envío a su destino, liberando automáticamente el pago hacia la empresa de transporte.

6.3.4 Control de Temperatura y Humedad

La aplicación de Smart Contracts garantiza condiciones ambientales específicas durante el transporte, crucial para sectores como la logística farmacéutica y de alimentos, donde mantener la temperatura y humedad adecuadas es fundamental.

En este caso, empresas de transporte pueden emplear sensores IoT para supervisar en tiempo real la temperatura y humedad en paquetes o contenedores. Los Smart Contracts podrían activarse cuando la señal GPS llegue al destino y los sensores confirmen que las condiciones adecuadas se mantuvieron durante el transporte. Si se cumplen las condiciones, el contrato inteligente podría liberar automáticamente el pago al transportista.

6.3.5 Detección de Exceso de Velocidad

Smart Contracts se implementan para monitorear y aplicar automáticamente límites de velocidad, promoviendo la seguridad vial y reduciendo los riesgos asociados con el exceso de velocidad.

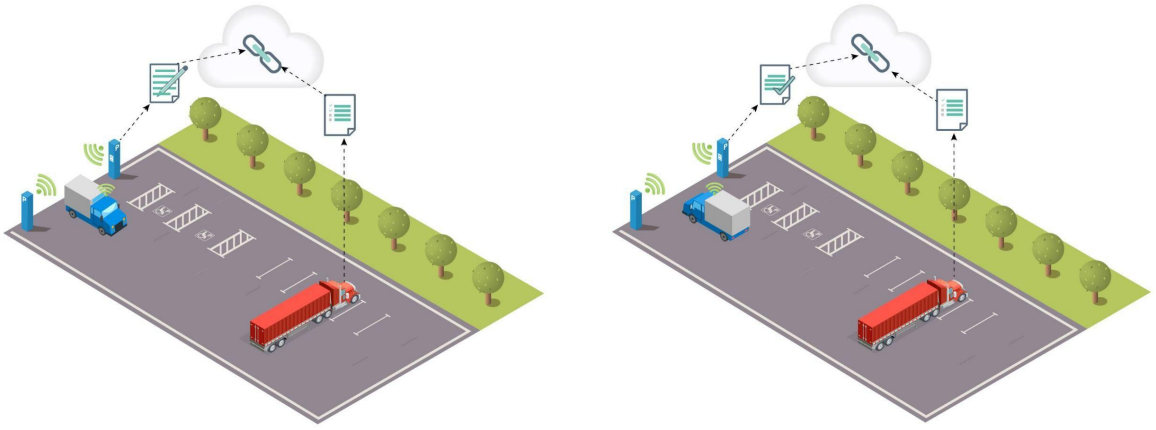
Retomando el ejemplo de la Empresa A en el primer caso, si una organización B (encargada del control de velocidad en la ciudad) también forma parte de la red blockchain, se podría generar un escenario donde un vehículo de la Empresa A exceda la velocidad permitida. Si los sensores de velocidad de la organización B detectan esta infracción, un Smart Contract podría activarse para imponer una multa por exceso de velocidad. El contrato retendría el costo de la multa de la Empresa A hasta que ésta valide la velocidad del vehículo en ese período. Si los valores coinciden, el contrato concluye, aprobando la multa y efectuando la transacción pagando la multa a la organización B.

6.3.6 Aparcamiento de vehículo por detección de movimiento

Mediante la integración de sensores IoT y Smart Contracts, se logra una gestión inteligente del estacionamiento, optimizando el espacio y reduciendo la congestión urbana.

Consideremos el escenario propuesto en el cual los vehículos requieren estacionarse en un área de estacionamiento en la playa o en las zonas autorizadas en la vía pública. Supongamos también que la Empresa A, dedicada al transporte, y la Organización B, encargada de la administración de estacionamientos, están integradas en una misma infraestructura blockchain. En este contexto, es posible implementar un contrato inteligente que supervise y facilite los procesos de pago asociados a los vehículos que se estacionan en el área designada. Mediante la utilización de un agente de detección de movimiento, el contrato inteligente se activa al detectar la entrada de un vehículo al área de estacionamiento. Desde ese momento, el contrato comienza a registrar el tiempo de estacionamiento del vehículo. Una vez que el vehículo se retira y la detección de movimiento confirma su salida, el contrato inteligente finaliza la sesión de estacionamiento, calculando con precisión el tiempo de ocupación. Como resultado, se realiza una transacción de pago desde la Empresa A hacia la Organización B, reflejando el período exacto de uso y garantizando una eficiencia en la transacción financiera, en la Figura 6.1.

Figura 6.1: Aparcamiento de vehículo por detección de movimiento



Este enfoque aprovecha la tecnología blockchain y los Smart Contracts para automatizar y optimizar el proceso de estacionamiento, reduciendo la necesidad de intervención humana y asegurando un registro preciso de las transacciones. Esta solución presenta un modelo eficiente y transparente para la gestión de estacionamientos, alineado con los avances tecnológicos y las necesidades modernas de movilidad urbana.

6.4 Resumen

En este capítulo, se examinaron diversas aplicaciones de Smart Contracts en entornos vehiculares respaldados por plataformas IoT. Desde la gestión eficiente de combustible hasta la supervisión predictiva del mantenimiento, estas aplicaciones ilustran cómo la integración de blockchain y IoT transforma el sector del transporte. Los Smart Contracts fueron implementados para monitorizar y optimizar los niveles de combustible, reduciendo pérdidas y mejorando la eficiencia operativa. Además, se exploraron casos de uso como el mantenimiento predictivo de dispositivos, la geolocalización en tiempo real de envíos, el control ambiental durante el transporte, la detección automática de exceso de velocidad y la optimización del estacionamiento mediante la combinación de sensores IoT y contratos inteligentes. Este análisis resalta cómo estas soluciones respaldadas por Smart Contracts

contribuyen a la eficiencia y seguridad en el transporte, marcando un paso significativo hacia sistemas más inteligentes y sostenibles en la era de la IoT y la blockchain.

Capítulo 7 Prototipo

7.1 Introducción

Este capítulo se adentra en la materialización práctica del agente de detección de movimiento para el estacionamiento vehicular, abordando aspectos cruciales como el modelado detallado del caso de uso mediante un Diagrama de Máquina de Estados y un Diagrama de Secuencia. Estas representaciones gráficas proporcionarán una comprensión profunda de cómo el agente responde a eventos específicos y cómo interactúa con otros elementos en el sistema, desde su activación hasta la conclusión del proceso.

Además, se presentará un Smart Contract en Solidity, la tecnología base que permite ejecutar automáticamente y de manera confiable las lógicas del contrato en una cadena de bloques. Este contrato inteligente encapsula las reglas y condiciones del estacionamiento, garantizando transparencia e inmutabilidad en cada transacción.

El capítulo también incluirá un enfoque práctico en el IDE de Remix, detallando cómo implementar y ejecutar este Smart Contract en un entorno de desarrollo interactivo. Se explorará cómo el contrato reacciona a los eventos del agente de detección de movimiento, permitiendo una visualización en tiempo real de cómo la blockchain procesa las transacciones.

Adicionalmente, se abordará el aspecto financiero del estacionamiento mediante la ejecución de un oráculo. Este oráculo proporcionará el costo del estacionamiento en tiempo real, integrando datos del mundo exterior en el contrato inteligente.

7.2 Modelos

En el contexto del agente de detección de movimiento para el estacionamiento vehicular, el Diagrama de Máquina de Estado se convierte en una herramienta esencial para comprender y representar cómo el agente responde a los diferentes estados a lo largo del tiempo. Este modelo será fundamental para visualizar claramente cómo el agente pasa de un estado a otro en respuesta a eventos específicos, como la detección de un vehículo ingresando o saliendo del área de estacionamiento. Además, permitirá identificar los diferentes estados del agente, desde el momento en que se activa hasta cuando se desactiva, proporcionando una visión detallada de su comportamiento dinámico.

El Diagrama de Secuencia, por otro lado, resulta crucial para representar las interacciones entre el agente de detección de movimiento, la plataforma IoT y el Contrato Inteligente en el proceso de estacionamiento vehicular. Este modelo visualizará claramente cómo se intercambian mensajes entre estos elementos a lo largo del tiempo, desde la activación del agente hasta la conclusión del contrato inteligente. En este caso específico, el Diagrama de Secuencia será valioso para comprender la secuencia de eventos que ocurren desde la detección del vehículo hasta la liberación de coins y la realización de la transacción de pago, proporcionando una visión detallada del flujo de ejecución en este escenario.

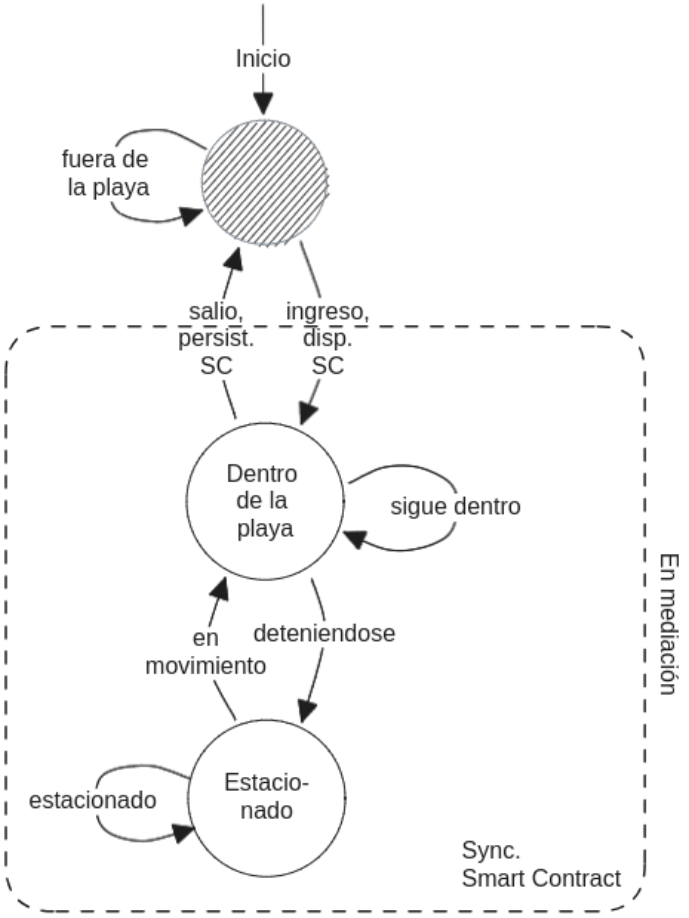
7.2.1 Aplicación de smart contract con agente de aparcamiento IoT

Siguiendo el proceso de la Figura 6.1 presentado en el capítulo anterior, específicamente el caso de estacionamiento de vehículos mediante detección de movimiento, se puede modelar de la siguiente manera, como se ilustra en la Figura 7.1. Comenzamos con el estado inicial en el que el vehículo se encuentra fuera del área de estacionamiento. La entidad a cargo debe ser capaz de identificar los vehículos que ingresan al área, lo cual se logra mediante tecnologías como RFID y sensores de detección de movimiento. Dado que

ambas organizaciones operan en la misma cadena de bloques, los sensores de movimiento y la identificación del vehículo activan el agente configurado en la plataforma IoT. Esto resulta en el despliegue de un SC que establece los valores iniciales.

El contrato verifica que el vehículo de la empresa de transporte pertenezca a la red y también captura los datos de la entidad que alquila el espacio de estacionamiento. Durante el tiempo en que el vehículo permanece en el área de estacionamiento, el estado del agente permanece constante. Estos estados se registran en el contrato inteligente para calcular el tiempo de permanencia del vehículo en el área. Simultáneamente, el contrato entra en un estado de mediación. A medida que transcurre el tiempo, las coins de la empresa de transporte se retienen con el propósito de asegurar el pago a la entidad encargada de administrar el estacionamiento. Cuando el vehículo se retira del área de estacionamiento y el estado del agente cambia a "fuera del establecimiento", el contrato inteligente valida los datos junto con la identificación RFID del vehículo. Esto concluye el contrato, liberando las coins y realizando la transacción de pago a la entidad responsable de administrar el espacio de estacionamiento.

Figura 7.1: Máquina de estado para el agente de detección de movimiento

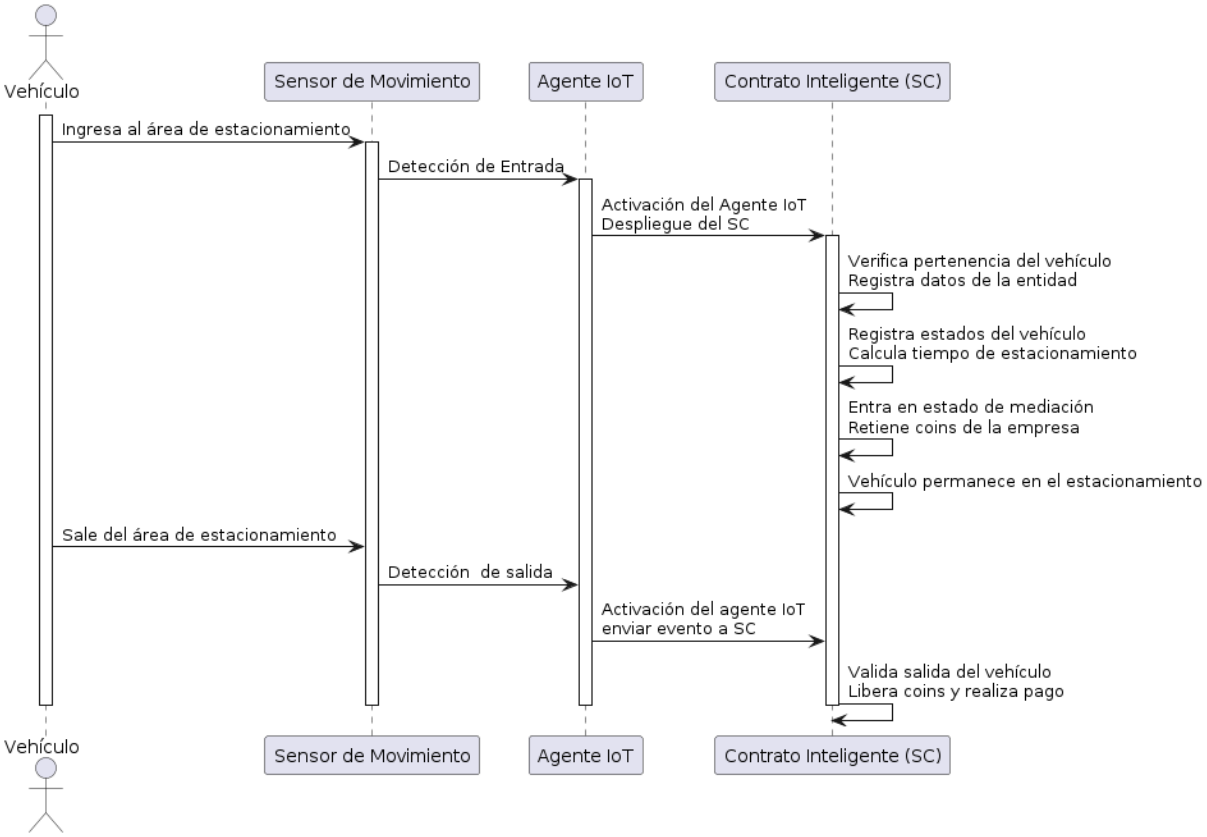


Significado de la mediación: la mediación en este contexto se refiere a un estado intermedio o de espera en el proceso. En el caso específico descrito, se trata del estado en el que el vehículo permanece dentro del área de estacionamiento y el contrato inteligente retiene los tokens como una medida de seguridad para garantizar el pago a la entidad que administra el espacio. Durante este período, el contrato está en un estado de "mediación" ya que está en espera y sigue monitoreando la situación hasta que se cumplan ciertas condiciones para proceder, como el retiro del vehículo y la validación de los datos necesarios.

A continuación en la Figura 7.2, el diagrama de secuencia modela el proceso de estacionamiento de vehículos mediante detección de movimiento, utilizando la tecnología de Smart Contracts en una plataforma IoT basada en blockchain. El diagrama sigue el flujo de

interacción entre los actores principales: el Vehículo, el Sensor de Movimiento, el Agente IoT y el SC.

Figura 7.2: Diagrama de secuencia para el agente de detección de movimiento



7.3 Parking Contract en Solidity

En la Figura 7.3, se muestra un programa simple para el caso del parking, está programado en Solidity, un lenguaje de alto nivel y orientado a objetos, que al compilarse corre sobre la EVM. En la Tabla 7.1 se muestran las variables que contiene el contrato.

Figura 7.3: Ejemplo simple de implementación SC Parking

```
// SPDX-License-Identifier: MIT
pragma solidity >=0.6.0;

import "@chainlink/contracts/src/v0.7/interfaces/AggregatorV3Interface.sol";

contract ParkingContract {
    address public owner; // Dueño del contrato
    address public parkingEntity; // Entidad que administra el estacionamiento
    uint256 public startTime; // Tiempo en que el vehículo ingresó al
estacionamiento
    uint256 public endTime; // Tiempo en que el vehículo salió del estacionamiento
    uint256 public paymentAmount; // Cantidad de tokens a pagar por el
estacionamiento

    enum State {
        Outside,
        Inside,
        Mediation,
        Completed
    }
    State public currentState;

    AggregatorV3Interface internal priceFeed;

    modifier onlyOwner() {
        require(msg.sender == owner, "Only owner can call this function");
        _;
    }

    constructor(
        address _parkingEntity,
        uint256 _paymentAmount,
        address _priceFeedAddress
    ) {
        owner = msg.sender;
        parkingEntity = _parkingEntity;
        paymentAmount = _paymentAmount;
        currentState = State.Outside;
        priceFeed = AggregatorV3Interface(_priceFeedAddress);
    }

    function enterParking() external {
        require(
            msg.sender == parkingEntity,
            "Only parking entity can call this function"
        );
        require(
            currentState == State.Outside,
            "Invalid state for entering parking"
        );
        startTime = block.timestamp;
        currentState = State.Inside;
    }

    function exitParking() external {
        require(
            msg.sender == parkingEntity,
            "Only parking entity can call this function"
        );
        require(
            currentState == State.Inside,
            "Invalid state for exiting parking"
        );
        endTime = block.timestamp;
        currentState = State.Mediation;
    }

    function validateExit(string memory rfid) external onlyOwner {
        require(
            currentState == State.Mediation,
            "Invalid state for validating exit"
        );
        require(
            keccak256(abi.encodePacked(rfid)) ==
                keccak256(abi.encodePacked("valid")),
            "Invalid RFID"
        );
        require(block.timestamp > endTime, "Vehicle still inside parking");
        currentState = State.Completed;
    }

    function getRemainingTime() public view returns (uint256) {
        require(
            currentState == State.Mediation,
            "Invalid state for getting remaining time"
        );
        return endTime - block.timestamp;
    }

    function withdrawPayment() external {
        require(currentState == State.Completed, "Payment not yet validated");
        require(
            msg.sender == parkingEntity,
            "Only parking entity can withdraw payment"
        );
        payable(parkingEntity).transfer(paymentAmount);
    }

    function getEthToUsdPrice() public view returns (int256) {
        (, int256 price, , , ) = priceFeed.latestRoundData();
        return price;
    }

    function calculatePaymentInUSD() public view returns (uint256) {
        int256 ethToUsdPrice = getEthToUsdPrice();
        return (paymentAmount * uint256(ethToUsdPrice)) / 1e8; // 1e8 is the
decimals of the price feed
    }
}
```

Estas líneas definen el contrato y declaran las variables que serán utilizadas en el contrato:

Aquí se define un modificador llamado **onlyOwner**, que permite restringir el acceso a ciertas funciones sólo al dueño del contrato. El constructor es la función que se ejecuta una vez al deployar el contrato. Establece al dueño del contrato, la entidad de estacionamiento y la cantidad de pago, además de poner el estado inicial del contrato como **Outside**.

enterParking y **exitParking** son funciones que la entidad de estacionamiento puede llamar para registrar la entrada y salida del vehículo. Estas funciones verifican que la entidad esté llamando la función en el estado correcto (**Outside** para **enterParking** y **Inside** para **exitParking**) y actualizan el estado y los tiempos correspondientes.

validateExit es una función solo para el dueño que valida la salida del vehículo. Verifica si el vehículo salió del estacionamiento en el tiempo correcto y con la RFID válida.

getRemainingTime permite a cualquiera verificar cuánto tiempo le queda al vehículo en el estacionamiento en el estado de mediación.

withdrawPayment permite a la entidad de estacionamiento retirar el pago después de que el vehículo haya salido y el pago haya sido validado. La cantidad de pago se transfiere a la entidad utilizando la función transfer.

getEthToUsdPrice que utiliza el oráculo Chainlink para obtener el precio actual de ETH en USD. La función **calculatePaymentInUSD** convierte el **paymentAmount** de ETH a USD utilizando el precio obtenido del oráculo.

AggregatorV3Interface es una interfaz proporcionada por Chainlink, un servicio de oracle para contratos inteligentes en blockchain. Esta interfaz está diseñada para interactuar con contratos de agregadores de precios, que son contratos específicos de Chainlink que proporcionan información sobre el precio de activos, como criptomonedas, en términos de otra moneda, como USD.

En el contexto de Ethereum y Chainlink, el **AggregatorV3Interface** generalmente incluye funciones estándar para obtener información sobre el precio actual de un activo en términos de otra moneda. Cuando se utiliza en un contrato inteligente, esta interfaz permite que el contrato consulte y utilice el precio actual de un activo externo en sus lógicas internas. En este caso, **priceFeed** es una variable de tipo **AggregatorV3Interface** que se utiliza para interactuar con el contrato de Chainlink que proporciona el precio de ETH en USD.

Tabla 7.1: Parking Contract variables

VARIABLES	DESCRIPCIÓN
owner y parkingEntity	Son las direcciones de Ethereum del dueño del contrato y de la entidad que administra el estacionamiento respectivamente.
startTime y endTime	Almacenan los momentos en que el vehículo entra y sale del estacionamiento.
paymentAmount	La cantidad de tokens que se deben pagar por el estacionamiento.
State	Es un enumerado que define los posibles estados del contrato: *Outside, Inside, Mediation y Completed*.
currentState	Almacena el estado actual del contrato
_priceFeedAddress	Es la dirección del contrato Chainlink Aggregator que proporciona el precio actual de ETH en USD. Los contratos Chainlink Aggregator son oráculos que proporcionan datos externos, como tasas de cambio, al contrato inteligente. Esta dirección se pasa al instanciar el contrato y se utiliza para acceder al oráculo Chainlink para obtener el precio de ETH en USD.
priceFeed	Es el nombre de la variable que representa la interfaz del contrato. Aquí, específicamente, se refiere al contrato que proporciona el precio actual de ETH en USD. La idea es que esta variable **priceFeed** sea inicializada con la dirección del contrato de Chainlink que proporciona el precio actual de ETH en USD. Luego, otras funciones del contrato pueden utilizar esta interfaz para obtener el precio actual y realizar cálculos en consecuencia.

7.4 Aplicaciones

El caso de uso de detección de movimiento para la gestión de estacionamientos mediante tecnologías IoT y contratos inteligentes presenta diversas aplicaciones prácticas en entornos urbanos y logísticos. Algunos casos de aplicabilidad incluyen:

1. **Estacionamientos Urbanos:**
    - *Control de Acceso:* Utilizando la detección de movimiento, se puede gestionar el acceso automatizado a estacionamientos urbanos, optimizando el flujo vehicular y mejorando la experiencia del usuario.
    - *Pago Automático:* Los contratos inteligentes permiten la automatización del proceso de pago basado en el tiempo real de ocupación, eliminando la necesidad de sistemas tradicionales de pago.
2. **Centros Logísticos:**
    - *Gestión de Flotas:* Integrando la detección de movimiento, se pueden monitorear y gestionar eficientemente las flotas de vehículos en centros logísticos, asegurando un estacionamiento adecuado y proporcionando datos en tiempo real sobre la disponibilidad.
3. **Seguridad Vehicular:**
    - *Prevención de Exceso de Velocidad:* Mediante la detección de movimiento, se pueden identificar vehículos que superan los límites de velocidad permitidos, contribuyendo a la seguridad vial en áreas de estacionamiento.
4. **Almacenes y Zonas de Carga:**
    - *Control de Movimiento en Áreas Específicas:* Para garantizar la eficiencia en zonas de carga y descarga, la detección de movimiento puede utilizarse para activar contratos inteligentes que regulen el tiempo y el acceso a estas áreas.
5. **Parques Empresariales:**

- *Gestión de Espacios de Estacionamiento Compartidos:* En entornos empresariales compartidos, la detección de movimiento y contratos inteligentes facilitan la asignación y pago de espacios de estacionamiento según la demanda.

Capítulo 8 Corroboración Empírica

8.1 Introducción

En el contexto de la investigación sobre la optimización de procesos contractuales en entornos organizacionales mediante la integración de SC y tecnologías IoT, se plantea la necesidad de realizar una corroboración empírica para validar y demostrar la eficacia del enfoque innovador.

Es importante señalar que, si bien existe la posibilidad de construir un entorno emulado para simular la interacción entre contratos inteligentes y dispositivos IoT, se optó por simplificar este proceso mediante el uso del IDE Remix para el despliegue del contrato. Esta elección se fundamenta en la conveniencia y accesibilidad que ofrece Remix, permitiéndonos centrarnos en la esencia de nuestra propuesta sin desviarnos del alcance de los temas específicos de la tesis.

En esta etapa de la investigación, se centra en la comprensión y validación de la ejecución precisa de contratos inteligentes en un entorno controlado, aprovechando la interfaz intuitiva y las funcionalidades proporcionadas por Remix. En lugar de emular un entorno completo con dispositivos IoT.

La elección de esta metodología permite enfocar la lógica y funcionalidades inherentes a la ejecución de contratos inteligentes en un contexto organizacional. En las secciones siguientes, se detalla un enfoque experimental y discutiremos los resultados obtenidos,

destacando el impacto potencial de la propuesta en la mejora de procesos contractuales y la eficiencia operativa en entornos empresariales.

Es importante señalar que la integración entre plataformas IoT y la red blockchain se puede lograr eficazmente a través de APIs, permitiendo una comunicación fluida entre estos dos entornos tecnológicos. Además, cabe destacar que aunque las plataformas blockchain ofrecen una capa adicional de seguridad y transparencia, no interfieren con el funcionamiento cotidiano de las plataformas IoT, sino que más bien, complementan y mejoran la gestión de los datos y procesos.

8.2 Remix - IDE nativo para el desarrollo de la Web3

Remix - Ethereum IDE

La elección del IDE Remix se fundamenta en su versatilidad y facilidad de uso para realizar pruebas empíricas del caso de uso del agente de detección de movimiento. Remix, como un IDE nativo para la Web3, proporciona una interfaz intuitiva y potente que permite desarrollar, desplegar y probar contratos inteligentes de manera eficiente. Su integración directa con la red Ethereum, su compatibilidad con múltiples versiones de Solidity y la posibilidad de ejecutar contratos en un entorno de prueba local o en la red de prueba de Ethereum, facilitan la validación del comportamiento del contrato inteligente en diversas condiciones. Además, Remix cuenta con un entorno de depuración que permite analizar y corregir posibles errores en el código, brindando una herramienta completa para la investigación y validación del caso de uso del agente de detección de movimiento en un entorno IoT basado en Ethereum.

Características Principales (Remix, n.d.):

- **Soporte para Solidity:** Remix ofrece un ambiente de desarrollo dedicado para Solidity, el lenguaje de programación principal para contratos inteligentes en la

plataforma Ethereum. Facilita la creación, prueba y despliegue de contratos de manera eficiente.

- **Interfaz Amigable:** La interfaz de Remix se destaca por su accesibilidad y facilidad de uso. Proporciona un espacio intuitivo donde los desarrolladores pueden escribir, depurar y probar sus contratos de manera eficiente.
- **Simulación y Pruebas**: Ofrece funcionalidades avanzadas para la simulación y prueba de contratos inteligentes antes de su implementación en la cadena de bloques. Esto contribuye a la detección temprana de posibles problemas y a la mejora de la calidad del código.
- **Conectividad con Redes Blockchain:** Remix permite la conexión con diversas redes blockchain, lo que facilita las pruebas en entornos específicos y proporciona un panorama más completo de cómo los contratos se comportarán en distintas condiciones.
- **Integración con Metamask:** Remix se integra de manera fluida con Metamask, una cartera de criptomonedas y puente a la Web3. Esta integración facilita la interacción con contratos desplegados y la gestión de transacciones.
- **Despliegue Sencillo:** Proporciona herramientas integradas para el despliegue directo de contratos inteligentes en la cadena de bloques Ethereum y otras redes compatibles.
- **Variedad de Plugins:** Remix admite una variedad de plugins que amplían sus funcionalidades, permitiendo a los desarrolladores personalizar su experiencia de desarrollo según sus necesidades específicas.

8.3 Prueba Empírica del Caso de Uso del Agente de Detección de Movimiento

La prueba empírica tiene como objetivo fundamental evaluar la funcionalidad y eficacia del SC desarrollado para el caso de uso del agente de detección de movimiento en entornos vehiculares. La validación se centra específicamente en el correcto despliegue y ejecución del SC en el entorno de desarrollo utilizando el IDE Remix. Esta prueba busca confirmar que el SC responde de manera coherente a las diferentes interacciones que podrían ocurrir durante el proceso de estacionamiento de vehículos, incluyendo la entrada y salida del vehículo, la validación de RFID, el cálculo del tiempo de permanencia y la gestión del pago. La verificación se realizará mediante la ejecución de funciones específicas del contrato inteligente en el entorno controlado de Remix.

Al alcanzar este propósito, se espera obtener información valiosa sobre el comportamiento del SC, identificar posibles problemas o limitaciones, y validar su adecuación para su implementación en entornos del mundo real. Además, esta prueba servirá como base para la evaluación de la aplicabilidad práctica del caso de uso del agente de detección de movimiento en el ámbito de los contratos inteligentes para la gestión de estacionamientos vehiculares.

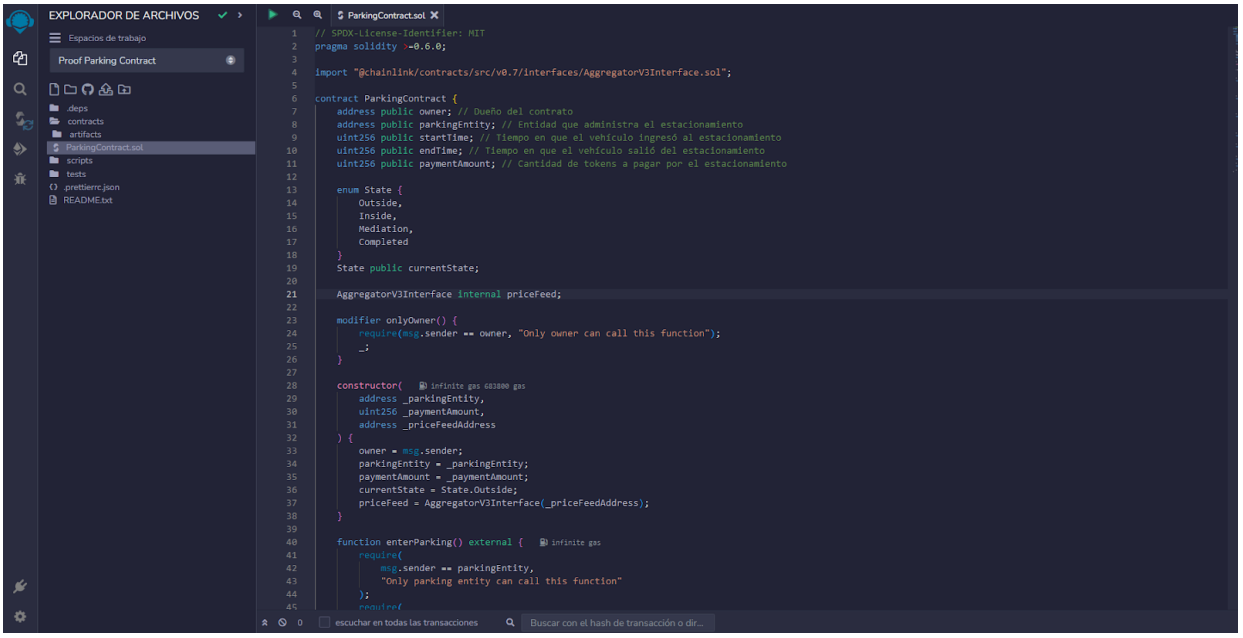
8.3.1 Configuración del Entorno

En el marco de la prueba empírica, se emplea un entorno de desarrollo centrado en el IDE Remix, que facilita la creación, despliegue y prueba de contratos inteligentes en la plataforma Ethereum. Además Remix posee funciones integradas que agilizan el proceso de desarrollo y pruebas.

La configuración específica del entorno se realiza considerando las características particulares de Remix VM:

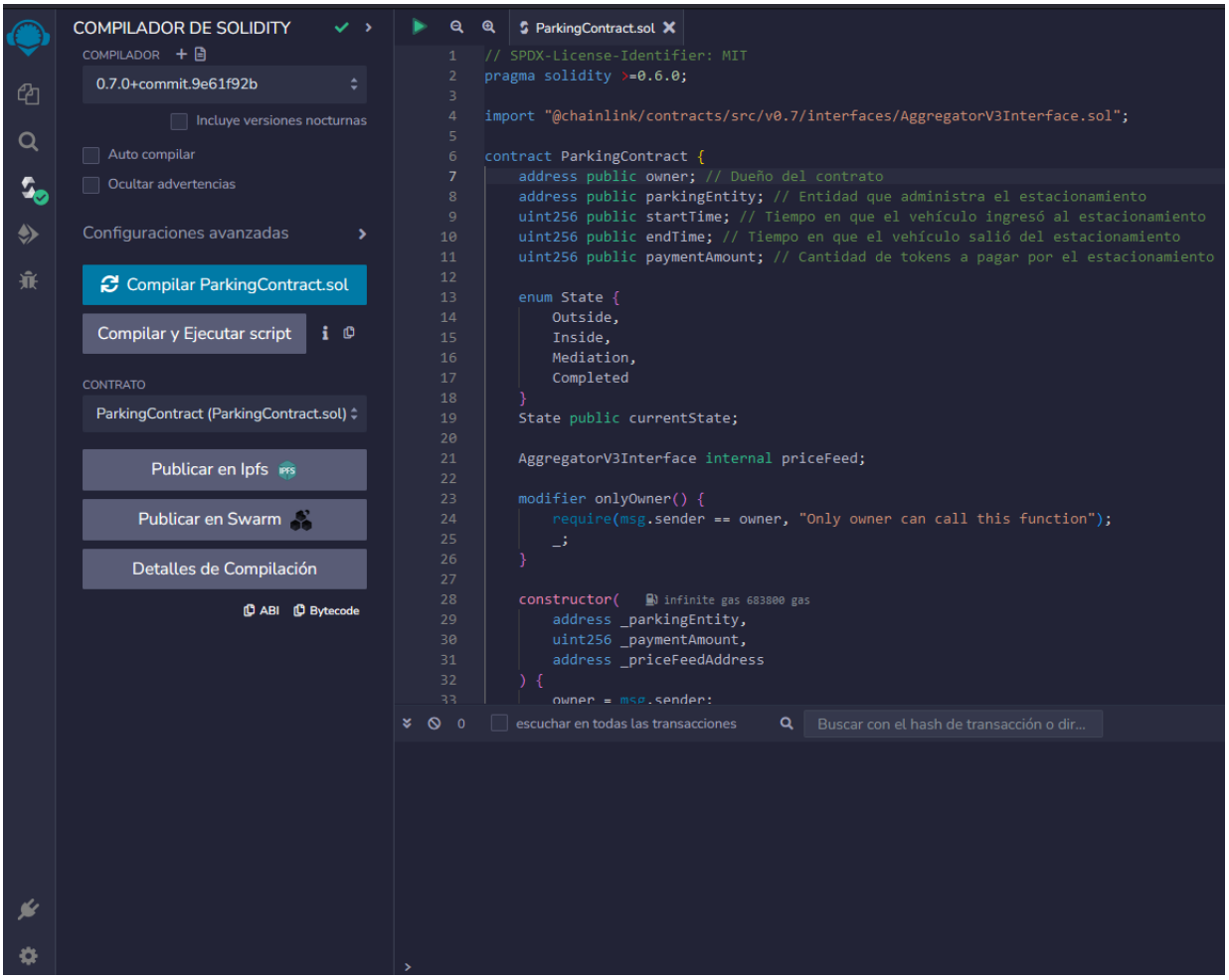
- **Inicio del Remix VM:** Se inicia Remix VM en el navegador web, lo que despliega un entorno de desarrollo local con funcionalidades completas de Remix, tal como se muestra el la Figura 8.1.

Figura 8.1: Inicio de Remix VM



- **Importación del Código Fuente:** El código fuente del SC se importa en Remix VM. Se verifican las dependencias y se realiza la compilación para asegurar la consistencia entre el código fuente y el bytecode generado, tal como se muestra el la Figura 8.2. Se puede verificar que dentro de la carpeta artifacts > build-info se encuentra varios archivos de compilación donde en el archivo ParkingContract.json se describen las especificaciones y el bytecode para que la EVM lo ejecute.

Figura 8.2: Compilación del código fuente



- **Configuración de Parámetros del Contrato:** Se ajustan los parámetros esenciales del SC, como la dirección de la entidad de estacionamiento y el monto de pago, para reflejar condiciones realistas y específicas del caso de uso, como se muestra en la Figura 8.4. En la Tabla 8.1 se especifican los datos.
- **Metamask**: Es una billetera de criptomonedas y una extensión para navegadores que simplifica la interacción con dApps en la red Ethereum. Permite gestionar activos digitales, interactuar con contratos inteligentes y cambiar entre redes blockchain, incluyendo redes de prueba como Goerli. En el contexto de Goerli, una red de prueba de Ethereum, Metamask facilita la realización de transacciones y pruebas de dApps en

un entorno de desarrollo seguro antes de implementar en la red principal. En la Figura 8.3 se muestra realizada la conexión a la red Goerli, con algunos eth ya adquiridos.

Figura 8.3: Metamask dashboard

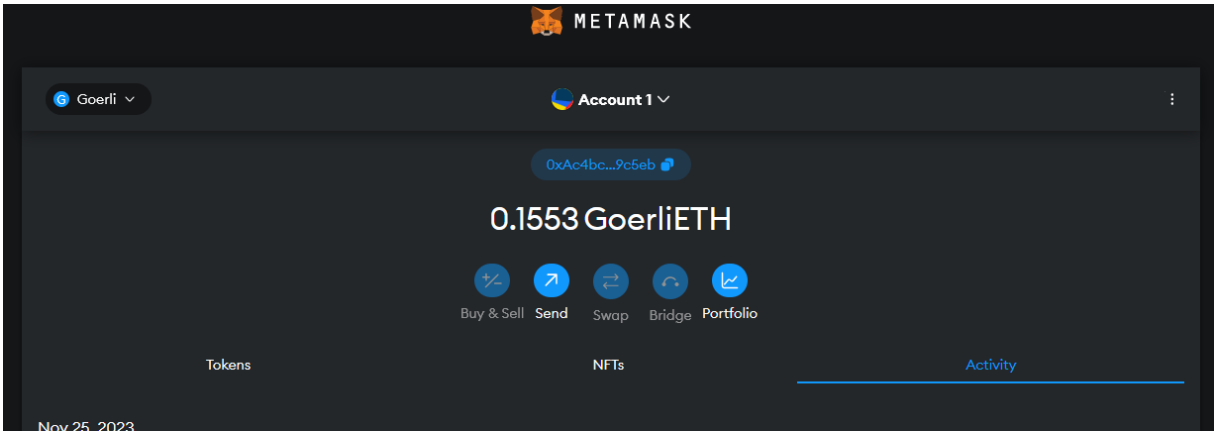
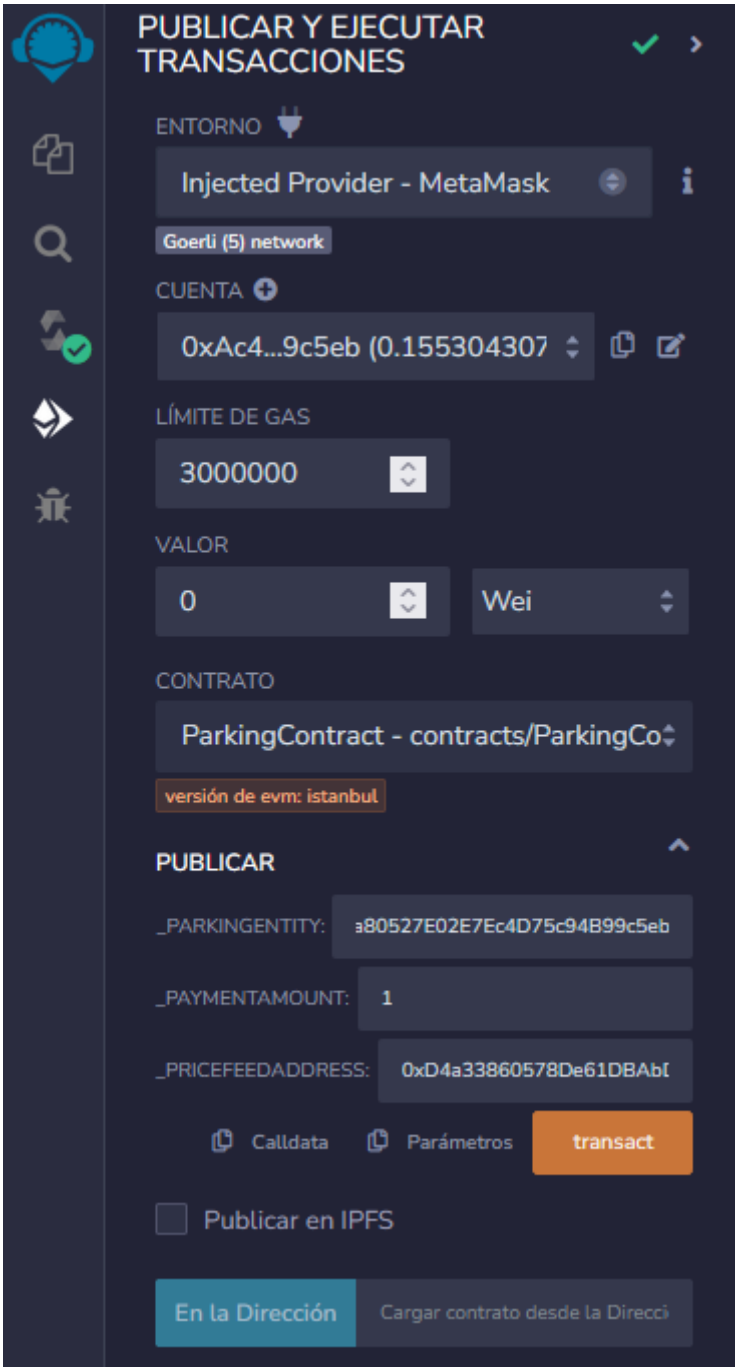


Tabla 8.1: Parámetros del Parking Contract

Variables iniciales	Cuenta	Ether
_PARKINGENTITY (Establecimiento)	0xAc4bc58eba453a80527E02E7Ec4D75 c94B99c5eb	100
_PRICEFEEDADDRESS	0xD4a33860578De61DBAbDc8BFdb98 FD742fA7028e	
_PAIMENTAMOUNT		1

El price feed address se puede obtener desde la documentación de Chainlink donde se encuentran las redes de prueba Goerli Testnet, en este caso se selecciona la relación ETH/USD.

Figura 8.4: Parámetros de ParkingContract en Remix



Este enfoque con Remix VM proporciona un ambiente de desarrollo local robusto y replicable, permitiendo llevar a cabo pruebas detalladas del SC del agente de detección de movimiento de manera eficiente y controlada.

8.3.2 Ejecución de Funciones del Smart Contract

La ejecución de funciones del Smart Contract en el IDE Remix se realiza mediante interacciones específicas que simulan acciones dentro del caso de uso del agente de detección de movimiento. Se despliega el contrato a través del botón transact y se ejecutan las funciones para realizar pruebas en este entorno.

Se ejecuta la función **enterParking,** se simula que entró el vehículo al área de estacionamiento, con lo cual se asigna el valor inicial del tiempo.

Luego se ejecuta la función **exitParking**, con lo cual se simula que el vehículo sale del área de estacionamiento, con lo cual se asigna el valor final del tiempo.

En la Figura 8.5, se muestra los resultados que se obtuvieron durante el contrato se mantuvo vigente, es decir, en mediación.

Se transforma la variable **startTime** que está en milisegundos a un timezone coherente Saturday, November 25, 2023 2:57:36 AM GMT-03:00

Idem con **endTime**, Saturday, November 25, 2023 2:58:36 AM GMT-03:00
Se observa que hubo una diferencia de 3 minutos.

currentState, tiene el valor 2, que corresponde a mediación. Esta variable se observó cuando estaba dentro de estos 3 minutos.

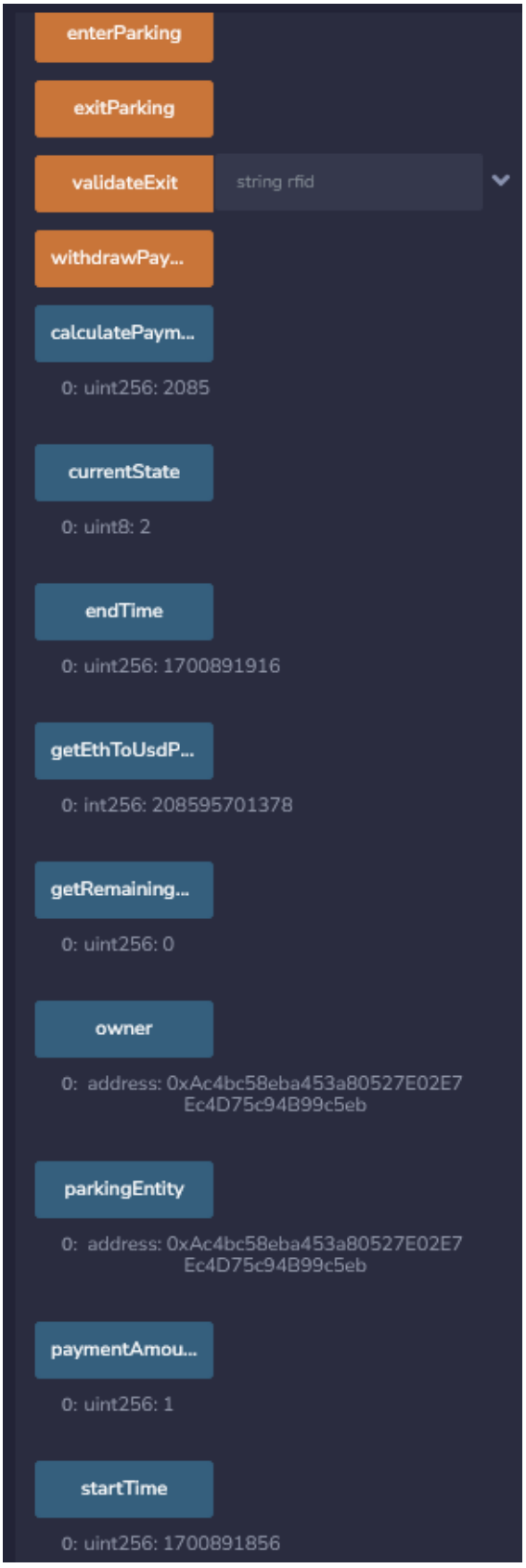
owner y parkingEntity, tiene la misma cuenta asociada, ya que es la que se asignó al desplegar el contrato.

Luego se tiene la función **getEthToUsdPrice** que devuelve el valor actual del precio del ETH, el valor es 208595701378 wei.

Por último, se tiene la función que calcula la conversión de ETH a USD **calculatePaymentInUSD,** que se basa en el oráculo de Chainlink de la red Goerli. Al pasarle 1 ETH en la variable **paymentAmount,** la función retorna el valor del ETH en USD, con el resultado de 2085 USD, que es en este momento el valor del ETH.

En este enlace se puede corroborar las transacciones y el código completo.

Figura 8.5: Instancia del contrato ParkingContract



Etherscan es un explorador de bloques para la red Ethereum. Funciona como una herramienta que permite a los usuarios explorar, analizar y seguir transacciones, direcciones y contratos en la cadena de bloques de Ethereum. Proporciona información detallada sobre bloques, transacciones, direcciones de Ethereum y contratos inteligentes. Los usuarios pueden utilizar Etherscan para obtener una visión transparente de la actividad en la red Ethereum, ver saldos de cuentas, y examinar el código fuente y la ejecución de contratos inteligentes.

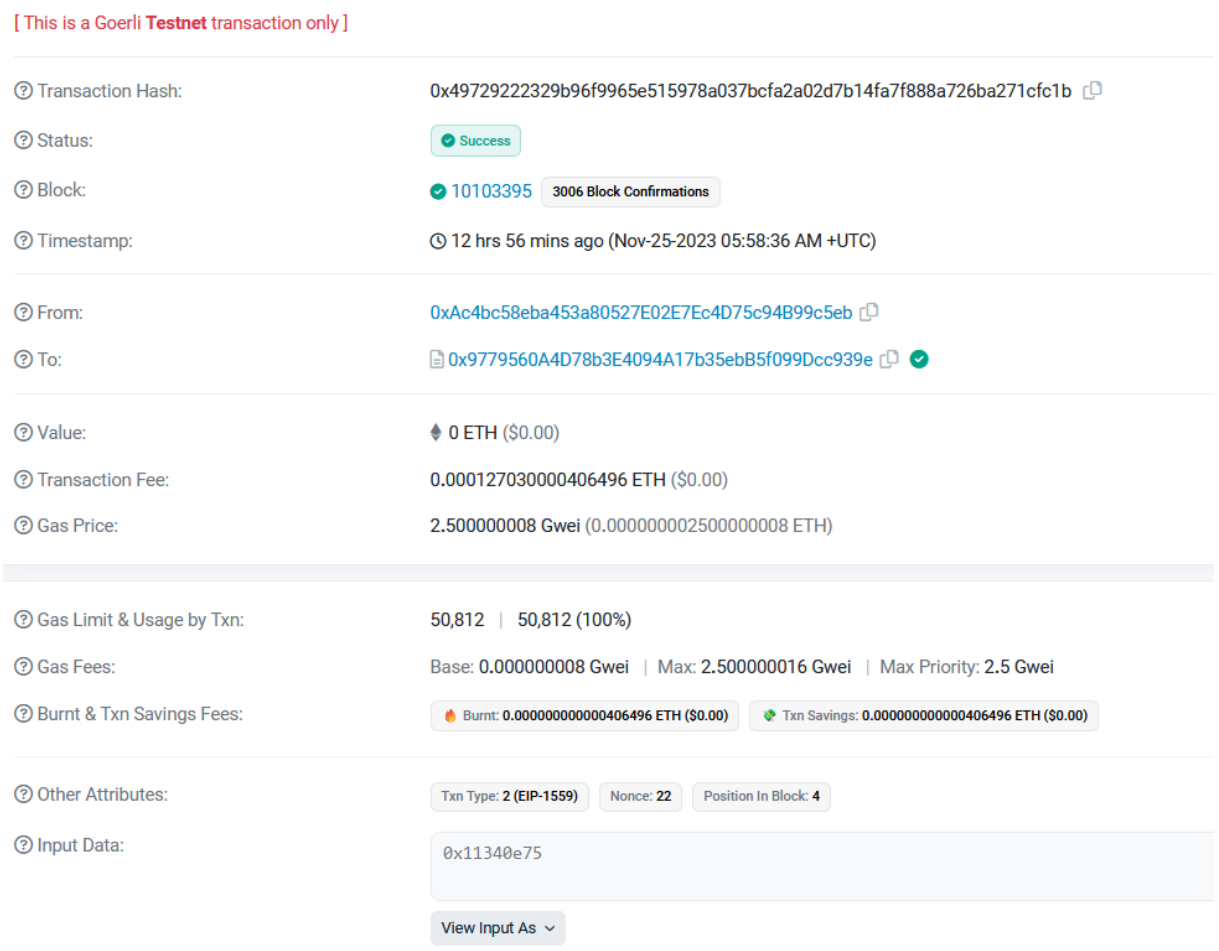
Analizando la última transacción en el Etherscan que se muestra en la Figura 8.6, se puede observar que en 25 de Noviembre de 2023 a las 05:58:36 AM +UTC, se realizó una transacción exitosa que involucra dos direcciones Ethereum. El remitente, con la dirección 0xAc4bc58eba453a80527E02E7Ec4D75c94B99c5eb, envió una cantidad de Ether sin valor específico (0 ETH) a la dirección del destinatario, 0x9779560A4D78b3E4094A17b35ebB5f099Dcc939e.

La transacción se incluyó en el bloque número 10103395 y ha recibido confirmación en 3006 bloques adicionales desde entonces. Esta operación fue realizada con éxito, y según la información del bloque, ocurrió alrededor de las 5:58 AM (hora UTC) del 25 de noviembre de 2023.

En términos de tarifas, la transacción no tiene un valor específico de Ether asignado y, por lo tanto, se muestra como $0.00. La tarifa de la transacción, conocida como "Gas Fee", fue de 0.00012703 ETH (aproximadamente $0.00). El precio del gas fue de 2.5 Gwei, y la transacción utilizó todo el gas asignado (50,812 unidades) al 100%.

Además, la información detalla las tarifas relacionadas con el gas, incluyendo el costo base, la tarifa máxima y la tarifa máxima de prioridad. También proporciona datos sobre el gas "quemado" durante la transacción y cualquier ahorro asociado.

Figura 8.6: Análisis del Transaction Hash



Esta transacción en Ethereum, aunque de bajo valor monetario, revela la eficiencia y transparencia inherentes a las operaciones en blockchain. La capacidad de rastrear cada aspecto de la transacción, desde el remitente hasta el destinatario, junto con los detalles específicos de las tarifas de gas, demuestra la auditabilidad y seguridad que ofrece la tecnología blockchain. Además, el hecho de que la transacción se haya completado con éxito, con una confirmación en 3006 bloques, subraya la robustez de la red Ethereum. Estos elementos destacan la utilidad y la confianza que pueden proporcionar las transacciones en blockchain, incluso en casos de bajo valor, y subrayan la importancia de la visibilidad y la trazabilidad en este entorno descentralizado.

8.4 Análisis y Recomendaciones

Para concebir y desarrollar una arquitectura de software en un entorno productivo centrado en las plataformas de IoT y la integración de la tecnología blockchain con contratos inteligentes relacionados con los eventos de estos sistemas, es fundamental abordar los desafíos inherentes. En primer plano, es crucial enfocarse en la tolerancia a fallos y garantizar una alta disponibilidad. Además, el sistema debe ser capaz de manejar eficazmente la carga de tráfico simultáneo y la concurrencia elevada, dado que un fallo en una transacción podría resultar en la pérdida de activos. Para contextualizar la magnitud de estos valores, consideremos dispositivos IoT que emiten señales en intervalos regulares. Por ejemplo, un GPS que emite señales cada dos segundos generaría 43200 señales en 24 horas. Sin embargo, esta cifra no considera la posibilidad de combinar múltiples dispositivos IoT, lo que aumentaría aún más el volumen de señales, sin mencionar también la cantidad de participantes en la red. La incorporación de una blockchain en una red preexistente para dar cabida a esta cantidad de señales se revelaría impracticable, dado los imperativos impuestos por los mecanismos de consenso requeridos. No obstante, si se lleva a cabo el desarrollo de una cadena de bloques ad hoc, se podría calibrar las propiedades centradas al entorno particular y solventar los obstáculos presentes en las plataformas IoT. En este entorno, se requeriría una arquitectura de alta complejidad, capaz de abordar de manera efectiva estos retos. Por ejemplo, se puede demostrar que los tipos de consensos tradicionales no son eficientes para las plataformas IoT (Huang et al., 2023). Por lo tanto, se propone un esquema de consenso supervisado basado en DPOS-PBFT mejorado (Delegated Proof of Stake-Practical Byzantine Fault Tolerance). El análisis y los resultados obtenidos sugieren que esta mejora podría ser una candidata prometedora para el desarrollo de Smart Contracts en sistemas IoT.

Por lo tanto, es esencial definir e identificar con precisión qué agentes serán los que desplegarán los contratos inteligentes. Esto permitirá minimizar tanto el costo computacional como el tiempo necesario para registrar los contratos en la blockchain. Además, esta selección estratégica contribuirá a reducir la congestión en la red, lo que a su vez disminuirá los costos asociados a las transacciones, proporcionando escalabilidad a largo plazo. Así, los contratos podrán ejecutarse automáticamente, llevando a cabo operaciones programadas de manera predecible en términos de tiempo y costos computacionales.

Conclusiones

Como se evidencia en esta investigación, resulta claro que las organizaciones están en un proceso constante de adopción de tecnologías innovadoras, realizando transformaciones digitales. En este contexto, el crecimiento constante del número de dispositivos conectados a Internet (IoT) y los avances en las tecnologías de comunicación están abriendo nuevas oportunidades. Específicamente, la combinación de la tecnología blockchain y la implementación de contratos inteligentes en plataformas IoT se presenta como una estrategia eficaz para mejorar la eficiencia de la gestión de datos y optimizar los procesos burocráticos. Esto resulta en una agilización de los flujos de trabajo y una mayor efectividad en el cumplimiento de contratos entre diferentes organizaciones.

Cabe resaltar que, aunque el enfoque principal de este estudio se dirigió hacia las plataformas de IoV, las implicaciones y resultados obtenidos son igualmente aplicables en Internet of Everything (IoE). La base conceptual y técnica implícita en la integración de la tecnología blockchain y los contratos inteligentes puede adaptarse a una amplia gama de contextos organizacionales, más allá de los vehículos.

El propósito esencial de este estudio es promover la adopción más generalizada de la tecnología blockchain en sistemas IoT. Esto permitiría un intercambio de datos más fluido y seguro entre distintas partes involucradas. Además, esta integración puede abordar de manera efectiva los desafíos asociados con la interoperabilidad, que es la capacidad de diferentes sistemas para comunicarse y colaborar de manera efectiva.

Al facilitar una gestión más eficiente de los datos y la automatización de los procesos, la tecnología blockchain puede contribuir significativamente a la promoción de la movilidad sostenible, un objetivo clave en un mundo cada vez más interconectado y consciente de su impacto ambiental.

Líneas Futuras de Investigación

Las líneas futuras de investigación identificadas abren la puerta a nuevas posibilidades, desde mejoras en la eficiencia del agente de detección de movimiento hasta la exploración de tecnologías emergentes y enfoques más avanzados en seguridad y privacidad. La búsqueda de la interoperabilidad entre cadenas de bloques y plataformas IoT, así como la consideración del impacto ambiental y sostenibilidad, refuerzan el compromiso con la innovación y la evolución constante de estas soluciones.

En última instancia, este trabajo sienta las bases para futuras investigaciones en el ámbito de la gestión inteligente del transporte, con un enfoque específico en la aplicación de tecnologías disruptivas como los Smart Contracts en entornos IoT. La convergencia de estas tecnologías promete transformar la manera en que interactuamos con el transporte y la logística, abriendo la puerta a un futuro más eficiente, seguro y sostenible.

A continuación se mencionan algunas líneas futuras de investigación que se podrían abordar.

Mejoras en la Eficiencia del Agente de Detección de Movimiento:

- Explorar tecnologías emergentes, como la inteligencia artificial y el aprendizaje profundo, para mejorar la precisión y capacidad predictiva del agente de detección de movimiento.
- Investigar el uso de sensores más avanzados y sistemas de visión computarizada para optimizar la identificación y seguimiento de vehículos en el área de estacionamiento.

Integración de Tecnologías Emergentes:

- Analizar la integración de tecnologías como el 5G para mejorar la conectividad y la transmisión de datos en tiempo real entre los agentes y la red blockchain.
- Explorar la aplicación de tecnologías edge computing para procesar datos de detección de movimiento de manera más eficiente y reducir la latencia.

Seguridad y Privacidad:

- Investigar en métodos avanzados de cifrado y técnicas de privacidad para proteger la información sensible recopilada por los agentes.
- Explorar la implementación de técnicas de privacidad basadas en blockchain para garantizar la integridad y confidencialidad de los datos.

Contratos Inteligentes Dinámicos:

- Desarrollar contratos inteligentes más dinámicos que puedan adaptarse a condiciones cambiantes, como tarifas de estacionamiento variables según la demanda o eventos específicos.
- Investigar el uso de oráculos más avanzados que permitan una mayor automatización y actualización en tiempo real de la información de tarifas.

Impacto Ambiental y Sostenibilidad:

- Evaluar el impacto ambiental de la implementación de los agentes y la ejecución de contratos inteligentes, buscando soluciones que minimicen la huella de carbono.

- Investigar iniciativas y prácticas sostenibles en el contexto de plataformas IoT y blockchain.

Anexo I

Historia de IoT

Este concepto fue acuñado en 1999 por Kevin Ashton, un empresario británico, durante su presentación a la firma Procter & Gamble. Sin embargo, la idea de dispositivos conectados surgió hace un par de décadas. Los primeros conceptos sobre la creación de una red de dispositivos inteligentes se discutieron en 1982, cuando una máquina de Coca-Cola modificada se convirtió en el primer electrodoméstico conectado a Internet. Desde entonces, la evolución de IoT ha permitido una mayor comprensión y una mejor eficiencia en distintos tipos de procesos empresariales (*Historia Y Evolución Del Internet De Las Cosas (IoT)* | *Tokio*, 2022).

Historia de la tecnología blockchain

En los primeros años de la década de 1990, se concibió la tecnología blockchain con el propósito de garantizar la integridad de los datos digitales y marcar su fecha y hora. Casi dos décadas después, Satoshi Nakamoto la adoptó para crear una criptomoneda, el Bitcoin. Así, el primer caso práctico de aplicación de la tecnología blockchain fue el Bitcoin.

Satoshi Nakamoto publicó en 2008 un artículo de investigación titulado "Bitcoin: un sistema de efectivo electrónico punto a punto", en el que postuló que las transacciones de esta criptomoneda podían ocurrir sin intermediarios. La llegada del Bitcoin en 2008 dio a conocer la tecnología blockchain, a pesar de que había sido concebida en los primeros años de la década de 1990. Pocos meses después, se presentó un nuevo protocolo que introdujo el concepto de un bloque génesis con 50 monedas. Este nuevo protocolo, que era de código abierto, se incorporó gradualmente a la red del Bitcoin (Islam et al., 2022).

Historia de Bitcoin

Bitcoin, la primera criptomoneda, fue creada en 2009 cuando Satoshi Nakamoto publicó el Libro Blanco de Bitcoin. El nacimiento de Bitcoin fue un proceso que va desde el origen de sus ideales y de la tecnología que le precedió, hasta más allá de la emisión del primer bitcoin (BTC). Esta moneda digital y su tecnología surgen como una respuesta a la necesidad de la gente por independizarse de los bancos centrales y convertirse en verdaderos dueños de su dinero. El movimiento cypherpunk jugó un papel protagónico en el desarrollo de Bitcoin. Los seguidores de esta causa fueron los que asentaron las bases filosóficas y tecnológicas de la precursora de todas las criptomonedas. Luego, Satoshi Nakamoto tomaría lo mejor de estas propuestas para dar origen a Bitcoin. El 3 de enero de 2009, se crea el primer bloque que genera 50 bitcoin y la primera transacción entre dos cuentas tiene lugar nueve días después. Al principio, un bitcoin costaba menos de $1 USD y llegó a costar casi $20,000 USD en diciembre del 2017. Bitcoin fue la primera criptomoneda en ser creada y que hace real la propuesta del dinero descentralizado y libre (Hughes & Gilmore, 2019).

Ethereum y contratos inteligentes

En 2013, el programador canadiense-ruso Vitalik Buterin, quien inicialmente contribuyó significativamente al código de Bitcoin, decidió crear otra criptomoneda conocida como Ethereum. Buterin percibió diversas limitaciones en la tecnología del Bitcoin, por lo que se propuso desarrollar una criptomoneda distinta. A diferencia del Bitcoin, que únicamente registra transacciones de su propia criptomoneda, el Ethereum permite a sus usuarios registrar datos digitales de elementos como automóviles, propiedades, yates y contratos (Islam et al., 2022).

El lanzamiento del Ethereum se concretó en 2015 y se caracteriza por su funcionalidad de contratos inteligentes. Estos contratos, basados en criterios predefinidos en la cadena de bloques, automatizan operaciones lógicas. Por ejemplo, un contrato inteligente puede emplearse para acordar una apuesta entre dos partes. Ambas partes depositarían su parte de la

apuesta en forma de criptomonedas y cargarían el contrato en la cadena de bloques del Ethereum. Las criptomonedas depositadas se administran a través del software subyacente de la cadena de bloques del Ethereum, y al concluir la apuesta, el contrato inteligente verificaría quién resulta ganador mediante operaciones lógicas y procedería a distribuir las ganancias al ganador (Islam et al., 2022).

Orígenes de la Web 3.0 en el Contexto de Blockchain

La Web 3.0 surge como respuesta a las limitaciones y desafíos inherentes a la Web 2.0, que se caracteriza por la centralización de datos y el control ejercido por grandes plataformas. Con la llegada de blockchain, una tecnología descentralizada, la visión de una internet más abierta y transparente comenzó a materializarse.

Blockchain, la tecnología subyacente en la Web 3.0, introdujo la noción de consenso distribuido, inmutabilidad y transparencia en el registro de datos. La descentralización inherente a blockchain permitió imaginar una web donde el control y la propiedad de los datos volvían a los usuarios. A medida que las capacidades de blockchain evolucionaron, la Web 3.0 se convirtió en un concepto que abraza la descentralización en todos los aspectos de la interacción en línea. La visión es crear un internet donde los usuarios tengan control sobre sus datos, las transacciones sean seguras y la censura sea prácticamente imposible.

Orígenes y Evolución de las dApps

Las Aplicaciones Descentralizadas (dApps) han surgido como un producto de la evolución constante en la tecnología blockchain. A medida que la comunidad blockchain buscaba extender sus capacidades más allá de las criptomonedas, las dApps se presentaron como una respuesta a la pregunta de cómo llevar la descentralización a diversas áreas de la vida digital.

El hito inicial y fundamental en el desarrollo de dApps se encuentra en el lanzamiento de Ethereum en 2015. Ethereum introdujo un concepto innovador: contratos inteligentes.

Estos contratos permitieron la creación de aplicaciones autoejecutables sin necesidad de intermediarios. La capacidad de programar acuerdos automáticos abrió la puerta a una nueva generación de aplicaciones descentralizadas. Las dApps tomaron impulso rápidamente, y su versatilidad pronto se hizo evidente. Se exploraron diversos sectores, desde finanzas hasta juegos y entretenimiento. La descentralización ofrecía no solo transparencia y seguridad, sino también la posibilidad de redefinir la propiedad y la participación en las plataformas digitales.

La introducción de NFTs llevó las dApps a nuevos horizontes, permitiendo la representación digital única y auténtica de activos. Esto abrió la puerta a la tokenización de obras de arte, bienes virtuales en juegos y otros elementos digitales, creando un mercado propio.A medida que más blockchains y protocolos se desarrollaban, las dApps comenzaron a explorar la interoperabilidad. La capacidad de interactuar entre diferentes plataformas se convirtió en un objetivo, dando lugar a proyectos que buscaban crear un ecosistema más conectado.

- **DeFi**: Las dApps en el sector DeFi han transformado los servicios financieros. Permiten transacciones directas, eliminando intermediarios y proporcionando acceso a una gama diversa de productos financieros, desde préstamos hasta intercambio de activos, de manera descentralizada.
- **Juegos y Entretenimiento**: El mundo del entretenimiento se ve revolucionado por dApps, especialmente en la industria de los juegos. La gamificación en blockchain introduce modelos de tokenización, NFTs y economías virtuales que proporcionan autenticidad y propiedad a los elementos del juego.
- **NFTs**: Las dApps basadas en NFTs han llevado la propiedad digital a un nuevo nivel. Desde obras de arte hasta bienes virtuales en juegos, los NFTs garantizan la autenticidad y singularidad de los activos digitales, permitiendo su compra, venta e intercambio en mercados descentralizados.

- **Contratos Inteligentes y Automatización**: Las dApps utilizan contratos inteligentes para automatizar procesos. Estos contratos autoejecutables, construidos en plataformas blockchain, garantizan transparencia y seguridad en diversas transacciones, desde acuerdos financieros hasta la gestión de activos.
- **Interoperabilidad y Desarrollo Continuo**: La interoperabilidad entre diferentes dApps y blockchains es un área clave de desarrollo. Proyectos como Polkadot y Cosmos buscan crear un ecosistema más conectado y versátil, permitiendo a las dApps colaborar y compartir información de manera eficiente.
- **Desarrolladores y la Evolución del Ecosistema:** Los desarrolladores de dApps desempeñan un papel central en la expansión y mejora continua del ecosistema. Utilizan lenguajes como Solidity para crear aplicaciones que aprovechan la descentralización, democratizando el acceso a servicios digitales.
- **Gamificación:** La integración de gamificación en dApp ofrece experiencias interactivas y atractivas. Juegos basados en blockchain utilizan contratos inteligentes para garantizar la transparencia y autenticidad de los elementos del juego, como activos y recompensas.

Blockchain Developer

Un Desarrollador Blockchain es un profesional especializado en la creación, diseño y mantenimiento de aplicaciones basadas en la tecnología blockchain. Este rol crucial implica comprender no solo los principios fundamentales de la tecnología blockchain, sino también la capacidad de aplicar esos conocimientos en la práctica. Las diferencias entre un Desarrollador Blockchain y el concepto de dApps radican en el enfoque y la amplitud de sus responsabilidades. Mientras que un Desarrollador Blockchain se centra en construir la infraestructura en sí de la cadena de bloques y sus componentes, una dApp se refiere a la aplicación específica construida sobre esta infraestructura.

Anexo II

En el marco de esta investigación, se adjuntan las evidencias de participación en el congreso 11° CoNaIISI 2023 que respaldan y enriquecen los fundamentos teóricos y prácticos abordados en el presente trabajo.

Las imágenes incluidas son evidencia tangible de la dedicación para mantenerse actualizado en las últimas tendencias, compartir conocimientos con la comunidad académica y establecer conexiones significativas con colegas e investigadores en el campo de estudio.

Se presenta a continuación un resumen del congreso en el cual se participó:

11° CONAIISI 2023

Congreso Nacional de Ingeniería Informática / Sistemas de Información

Área: Aplicaciones Informáticas y Sistemas de Información

Categoría: Profesional/Docente/Investigador

Fecha: 02/11/2023

Tema: Método de Optimización en Procesos Contractuales Interorganizacionales utilizando Smart Contracts en Plataformas IoT

Certificado

Que el trabajo: ***Método de optimización en procesos contractuales interorganizacionales utilizando Smart Contracts en plataformas IoT - Categoría Docentes-Investigadores****, autores:* ***Paulo José Ordóñez Giovanazzi - Iván Manuel Ordóñez Giovanazzi - Ulises Gabriel Pignatelli - Alejandro Mario Hernández*** *fue aprobado para ser presentado en el* ***11° Congreso Nacional de Ingeniería Informática / Sistemas de Información- CoNaIISI 2023,*** *organizado por la Red RIISIC perteneciente al CONFEDI y el Departamento de Ingeniería en Sistemas de Información de la Universidad Tecnológica Nacional - Facultad Regional Tucumán, realizado en modalidad hibrida los días 2 y 3 de noviembre de 2023 en la Facultad Regional Tucumán de la Universidad Tecnológica Nacional.*

San Miguel de Tucumán, 25 de noviembre de 2023

Mg. Ing. Gastón Martin
Coordinador 2023 - RIISIC

Lic. Augusto José Nasrallah
Coordinador CONAIISI2023

Mg. Ing. Walter Fabian Soria
Decano FRT - UTN

Certificado disertante

Que el trabajo: ***Método de optimización en procesos contractuales interorganizacionales utilizando Smart Contracts en plataformas IoT - Categoría Docentes-Investigadores,*** *fue presentado por:* ***Paulo José Ordóñez Giovanazzi*** *en el* ***11° Congreso Nacional de Ingeniería Informática / Sistemas de Información- CoNaIISI 2023,*** *organizado por la Red RIISIC perteneciente al CONFEDI y el Departamento de Ingeniería en Sistemas de Información de la Universidad Tecnológica Nacional - Facultad Regional Tucumán, realizado en modalidad hibrida los días 2 y 3 de noviembre de 2023 en la Facultad Regional Tucumán de la Universidad Tecnológica Nacional.*

San Miguel de Tucumán, 27 de noviembre de 2023

Mg. Ing. Gastón Martin
Coordinador 2023 - RIISIC

Lic. Augusto José Nasrallah
Coordinador CONAIISI2023

Mg. Ing. Walter Fabian Soria
Decano FRT - UTN

aplic_informaticas conaiisi2023

To: Sr Paulo José Ordóñez Giovanazzi <paulo.go@proton.me>

Wed 10/11/2023 10:13 AM

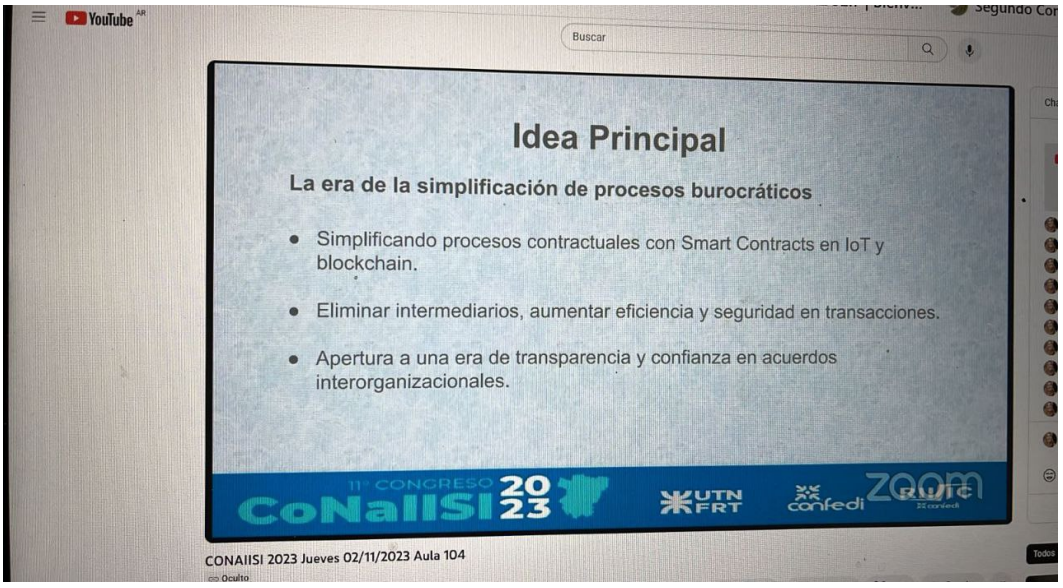
Cc: Ord��ez Giovanazzi, Paulo Jos�; Ord��ez, Iv�n Manuel; Pignatelli, Ulises Gabriel; Alejandro Mario Hernandez <alejandromario.hernandez@uai.edu.ar>

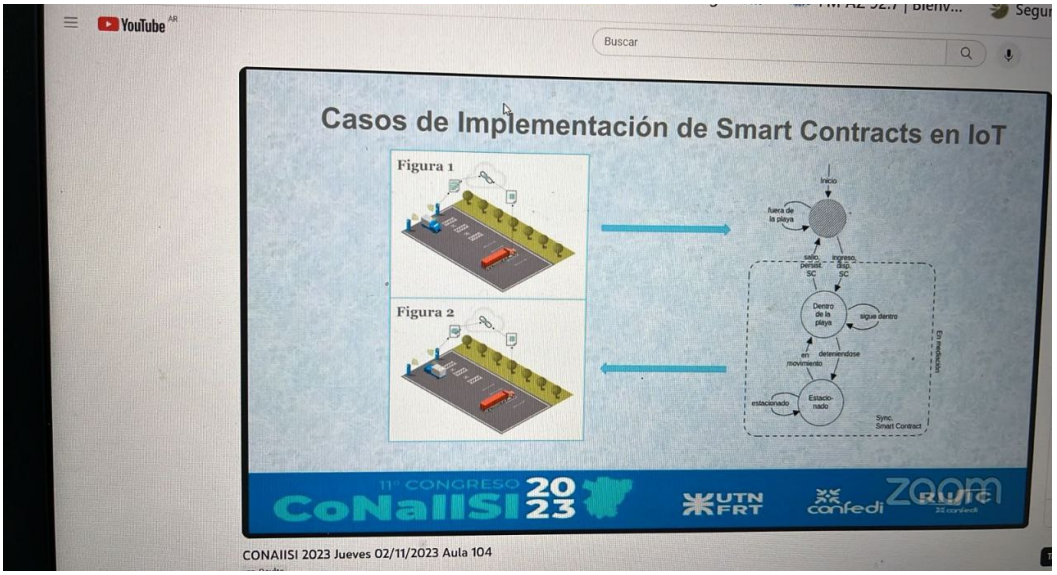
Sr Paulo José Ordóñez Giovanazzi:

Felicitaciones!!!
Tenemos el agrado de informarle que ha finalizado la revisión de su envío con el título "Método de optimización en procesos contractuales interorganizacionales utilizando Smart Contracts en plataformas IoT" y que él mismo ha sido aceptado para ser presentado en la onceava edición del CoNAIISI, que tendrá lugar los días 2 y 3 de Noviembre de 2023 en la ciudad de San Miguel de Tucumán, organizado por el Departamento Sistemas de Información la Facultad Regional Tucumán - Universidad Tecnológica Nacional, Argentina.

PROGRAMA de ACTIVIDADES – CoNaIISI2023

JUEVES 2 de noviembre						
8:00a 12:00hs.	Hall de Entrada de UTN - FRT		ACREDITACIÓN			
9:00a 10:00hs.	Aula Magna		ACTODEAPERTURA– 11°CoNaIISI			
10:00a11:00hs.	Aula Magna		CONFERENCIA INAUGURAL: "PhaseIDentificationSystem: Machine Learning para uso estratégico de Medidores Inteligentes" Ph.DAdrian Will			
11:00a11:30hs.	COFFEE BREAK					
11:30a13:00hs.	SESION 1– Jueves 2 de Noviembre					
AULA/AREA	HORA INICIO	ID	TÍTULO	AUTORES	FILIACIÓN	MODALIDAD
Aula 104 Aplicaciones Informáticas y de Sistemas de Información	14:30	210	Aplicación Mobile Learning con reconocimiento facial para la Enseñanza de Matemática a Niños con Necesidades Educativas Especiales en Contextos de Aulas Inclusivas	Agustín ,Alvarez Ferrando	Facultad Regional La Plata - Universidad Tecnológica Nacional	Presencial
	14:50	67	Módulo Gateway de Bluetooth para emulador de STM32 dentro de Docker	Esteban ,Carnuccio - Mariano Volker - Matías , Adagio - Dario , Hirschfeldt - Andrea , Vera	Universidad Nacional de La Matanza	Virtual
	15:10	69	Método de optimización en procesos contractuales interorganizacionales utilizando Smart Contracts en plataformas IoT	Paulo José, Ordóñez Giovanazzi - Iván Manuel Ordóñez Giovanazzi - Ulises Gabriel, Pignatelli - Alejandro Mario, Hernandez	Universidad Abierta Interamericana. Facultad de Tecnología Informática.	Virtual
	15:30	181	Uso de aprendizaje profundo para la localización de regiones de interés en imágenes médicas	Marcelo , Cappelletti - Christian Botta - Lucas , Olivera - Martín , Morales	TICAPPS-Universidad Nacional Arturo Jauretche, Florencio Varela, Argentina. - GCA, Instituto LEICI (UNLP-CONICET)	Virtual
Aula 108 Base de Datos	14:30	310	SVM Maquinas de vectores de Soporte para predecir la utilización del aire en la climatización de una Industria Alimenticia	Reinaldo David, Gomez	Universidad Nacional de Villa Mercedes	Presencial
	14:50	156	Proceso KDD extendido a Data Streams	Nélida Raquel, Cáceres	Facultad de Ingeniería- Universidad Nacional de Jujuy	Virtual
	15:10	218	Exploración de Relatos Personales a través de la Minería de Textos y Análisis de Sentimientos	Soledad , Ruiz Diaz - Miguel Mendez-Garabetti	Departamento de Posgrados, Universidad CAECE. Free and Open Source Software/Hardware ResearchLaboratory (FOSSHLab), Argentina	Virtual
	15:30	284	Indicadores cienciométricos preliminares sobre la investigación estudiantil en CoNaIISI	Roberto Miguel, Muñoz - Martín Gustavo Casatti - Analía , Guzmán - Juan Carlos, Cuevas	Universidad Tecnológica Nacional- Facultad Regional Córdoba -	Virtual
Aula SUM 1	14:30 a 16:00	**Comisión Vinculación – Panel After Emprendedor** **"Experiencias y Tendencias de Emprender en Tecnología"** **Panelistas: Ing. Catalina Mamani, Ivan -tadei, Ing. Gabriel Cerrutti, Mag. Ing. Gastón Martín** **Moderador: Esp. Ing. Valeria Poliche**				Presencial


YouTube
Buscar
Idea Principal
La era de la simplificación de procesos burocráticos
Simplificando procesos contractuales con Smart Contracts en IoT y blockchain.
Eliminar intermediarios, aumentar eficiencia y seguridad en transacciones.
Apertura a una era de transparencia y confianza en acuerdos interorganizacionales.
11° CONGRESO CoNaIISI 2023
UTN FRT
confedi
RIITIC
zoom
CONAIISI 2023 Jueves 02/11/2023 Aula 104


YouTube
Buscar
Casos de Implementación de Smart Contracts en IoT
Figura 1
Figura 2
Inicio
fuera de la playa
Dentro de la playa
sigue dentro
en movimiento
deteniendose
Estacionado
estacionado
En medición
Sync. Smart Contract
11° CONGRESO CoNaIISI 2023
UTN FRT
confedi
RIITIC
zoom
CONAIISI 2023 Jueves 02/11/2023 Aula 104


YouTube
Buscar
11° CONGRESO CoNaIISI 2023
UTN FRT
confedi
RIITIC
UTN
69-Método de Optimización en Procesos Contractuales Interorganizacionales
utilizando Smart Contracts en Plataformas IoT
Paulo Ordóñez Giovanazzi, Iván Ordóñez Giovanazzi, Alejandro Hernandez, Ulises Pignatelli y Universidad Abierta Interamericana. Facultadde Tecnología Informática. Centro de Altos
Estudios en Tecnología Informática. Rosario, Argentina.
paulojose.ordonezgiovanazzi@alumnos.uai.edu.ar
ivanmanuel.ordonez@alumnos.uai.edu.ar
San Miguel de Tucumán, Tucumán, Argentina – 2 y 3 de Noviembre 2023
zoom
CONAIISI 2023 Jueves 02/11/2023 Aula 104

Acrónimos

Acrónimo	Significado
5G	Red de Generación 5
AMQP	Advanced Message Queuing Protocol
API	Application Programming Interface
BIoV	Blockchain Internet de Vehículos
BTC	Bitcoin
CoAP	Constrained Application Protocol
CO2	Dióxido de Carbono
CS	Cadena de Suministro
DDoS	Distributed Denial of Service
DoS	Denial of Service
dApps	Decentralized Applications
EVM	Ethereum Virtual Machine
GCP	Google Cloud Platform
GPS	Global Positioning System

HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IA	Inteligencia Artificial
IDE	Integrated Development Environment
IETF	Internet Engineering Task Force
IoT	Internet of Things
IoV	Internet of Vehicles
ITS	Intelligent Transportation System
LoRaWAN	Long Range Wide Area Network
LPWAN	Low Power Wide Area Network
M2M	Machine to Machine
ML	Machine Learning
MQTT	Message Queuing Telemetry Transport
NFC	Near Field Communication
NFTs	Non-Fungible Tokens
P2P	Peer-to-Peer

PBFT	Practical Byzantine Fault Tolerance
PDF	Portable Document Format
PoS	Proof of Stake
PoW	Proof of Work
RFID	Radio Frequency Identification
SC	Smart Contract
V2I	Vehicle to Infrastructure
V2N	Vehicle to Network
V2P	Vehicle to Pedestrian
V2V	Vehicle to Vehicle
VANETs	Vehicular Ad Hoc Networks
VM	Virtual Machine
Wi-Fi	Wireless Fidelity

Referencias

Åkerberg, J., Gidlund, M., & Bjorkman, M. (2011). Future research challenges in wireless sensor and actuator networks targeting industrial automation. *IEEE International Conference on Industrial Informatics (INDIN)*. https://doi.org/10.1109/INDIN.2011.6034912

Ali, J., Ali, T., Musa, S., & Zahrani, A. (2018). Towards secure IoT communication with smart contracts in a Blockchain infrastructure. *International Journal of Advanced Computer Science and Applications*, *9*. 10.14569/IJACSA.2018.091070

Ashraf, M., & Heavey, C. (2023). A Prototype of Supply Chain Traceability using Solana as blockchain and IoT. *Procedia Computer Science, 217,* 948-959. 10.1016/j.procs.2022.12.292

Boncea, R., Petre, I., & Vevera, A. V. (2019, April). Building trust among things in omniscient Internet using Blockchain Technology. *Romanian Cyber Security Journal*. https://www.researchgate.net/publication/336284645_Building_trust_among_things_in_omniscient_Internet_using_Blockchain_Technology

Christidis, K., & Devetsikiotis, M. (2016, May 10). Blockchains and Smart Contracts for the Internet of Things. *IEEE Access*, *4*(16042927), 2292 - 2303. 10.1109/ACCESS.2016.2566339

Connected car: what is it and what is its future? (2022, November 26). Telefónica. Retrieved November 12, 2023, from https://www.telefonica.com/en/communication-room/blog/connected-car-what-is-it-and-what-is-its-future/

Dhakal, A., & Cui, X. (2018, April). Blockchain and Smart Contracts for Internet of Things: A Systematic Literature Review. *International School of Software, Wuhan University,*

China. Retrieved 06 03, 2023, from https://www.researchgate.net/publication/332671231_Blockchain_and_Smart_Contracts_for_Internet_of_Things_A_Systematic_Literature_Review

Ericsson Mobility Report November 2022. (2022). *Ericsson*, 11-15. https://www.ericsson.com/4ae28d/assets/local/reports-papers/mobility-report/documents/2022/ericsson-mobility-report-november-2022.pdf

Ethereum Homestead. (n.d.). ethereum.org. Retrieved November 15, 2023, from https://ethereum.org/en/

The future of Intelligent Transport Systems (ITS) | Engaged IT for the CIO. (2011, November 29). Engaged IT for the CIO. Retrieved November 16, 2023, from https://mubbisherahmed.wordpress.com/2011/11/29/the-future-of-intelligent-transport-systems-its/

Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013, September). Future Generation Computer Systems Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, *29*, 1645-1660. https://doi.org/10.1016/j.future.2013.01.010

Historia y evolución del Internet de las Cosas (IoT) | Tokio. (2022, April 19). Tokio School. Retrieved November 21, 2023, from https://www.tokioschool.com/noticias/internet-de-las-cosas-evolucion/

Huang, R., Yang, X., & Ajay, P. (2023, March 3). Consensus mechanism for software-defined blockchain in internet of things. *Internet of Things and Cyber-Physical Systems*, *3*, 52-60. doi.org/10.1016/j.iotcps.2022.12.004

Hughes, E., & Gilmore, J. (2019, July 2). *Historia y Trayectoria de Bitcoin*. Bitcoin Mexico. Retrieved November 21, 2023, from https://www.bitcoin.com.mx/infografia-historia-y-trayectoria-de-bitcoin/

Islam, S. H., Pal, A. K., Samanta, D., & Bhattacharyya, S. (Eds.). (2022). *Blockchain Technology for Emerging Applications: A Comprehensive Approach*. Elsevier Science. https://doi.org/10.1016/C2020-0-03280-9

JABBAR, R., DHIB, E., BEN SAID, A., KRICHEN, M., (Member, IEEE), FETAIS, N., (Senior Member, IEEE), ZAIDAN, E., & KAMEL BARKAOUI, A. (2022, February 07). Blockchain Technology for Intelligent Transportation Systems: A Systematic Literature Review. *IEEE VEHICULAR TECHNOLOGY SOCIETY SECTION*. 10.1109/ACCESS.2022.3149958

Jain, S. (2023, February 26). *Smart Contracts vs Traditional Contracts - What are the differences?* GeeksforGeeks. Retrieved November 12, 2023, from https://www.geeksforgeeks.org/smart-contracts-vs-traditional-contracts/

Kim, Y., Oh, H., & Kang, S. (2017). Proof of Concept of Home IoT Connected Vehicles. *Sensors*, *17*, 1289. 10.3390/s17061289

Kumar Sadhu, P., Yanambaka, V. P., & Abdelgawad, A. (2022, September 30). Internet of Things: Security and Solutions Survey. *Sensors 2022, 22(19)*(7433). doi.org/10.3390/s22197433

Kumar Sadhu, P., Yanambaka, V. P., & Abdelgawad, A. (2022, September 30). Internet of Things: Security and Solutions Survey. *Sensors 2022, 22(19)*(7433). doi.org/10.3390/s22197433

KUSHWAHA, S. S., JOSHI, S., SINGH, D., KAUR, M., & LEE, H.-N. (2022, January 4). Systematic Review of Security Vulnerabilities in Ethereum Blockchain Smart Contract. *IEEE*. 10.1109/ACCESS.2021.3140091

Mikolajczyk, M., & Ordaz, M. (n.d.). *Motoro: The Future of IoT in Transportation*. Toptal. Retrieved July 17, 2023, from https://www.toptal.com/ethereum/motoro-iot-in-transportation

Ozyilmaz, K. R., Dogan, M., & Yurdakul, A. (2018, June 20 - 22). IDMoB: IoT Data Marketplace on Blockchain. *2018 Crypto Valley Conference on Blockchain Technology*. https://dx.doi.org/10.1109/CVCBT.2018.00007

Paricherla, M., Babu, S., Phasinam, K., Pallathadka, H., Zamani, A. S., Narayan, V., Shukla, S. K., & Mohammed, H. S. (2022, May 17). Towards Development of Machine Learning Framework for Enhancing Security in Internet of Things. *Mukesh Soni, 2022*. doi.org/10.1155/2022/4477507

R, A. (2022, June 4). *Smart Home Solution using Smart Contract | Full DIY Project*. Electronics For You. Retrieved July 17, 2023, from https://www.electronicsforu.com/electronics-projects/hardware-diy/smart-home-solution-using-smart-contract

Rauch, S. (2021, April 12). *Leveraging Smart Contracts for IoT Applications*. Simplilearn.com. Retrieved November 8, 2023, from https://www.simplilearn.com/leveraging-smart-contracts-for-iot-applications-article

Remix. (n.d.). Remix - Ethereum IDE & community. Retrieved November 23, 2023, from https://remix-project.org/

Saad, M., Khalid Khan, M., & Bin Ahmad, M. (2022, March 25). Blockchain-Enabled Vehicular Ad Hoc Networks: A Systematic Literature Review. *Karachi Institute of Economics and Technology (KIET), College of Computing and Information Sciences (CoCIS), Karachi 75190, Pakistan*. https://doi.org/10.3390/su14073919

Solidity Programming Language. (n.d.). Home | Solidity Programming Language. Retrieved November 15, 2023, from https://soliditylang.org/

Taherdoost, H. (2023, February 13). Smart Contracts in Blockchain Technology: A Critical Review. (117). doi.org/10.3390/info14020117

Ulrich, A. (n.d.). *Las unidades del ether de Ethereum — De idiomas y números*. Of Languages and Numbers. Retrieved November 15, 2023, from https://www.languagesandnumbers.com/articulos/es/ethereum-ether-unidades/

United Nations (Ed.). (2021). *UNECE Nexus: Sustainable Mobility and Smart Connectivity*. UN.