UNIVERSIDAD DE OVIEDO

DEPARTAMENTO DE INFORMÁTICA

TESIS DOCTORAL

META-MODELO DE CONTRATOS INTELIGENTES USANDO CADENAS DE BLOQUES APLICADO AL SECTOR PÚBLICO

Meta-model of smart contracts using Blockchain applies to the public sector

AUTOR

Jenny Alexandra Triana Casallas

DIRECTOR

Dr. Juan Manuel Cueva Lovelle

Oviedo, octubre de 2021

RESUMEN

La última década, se ha caracterizado por un desarrollo importante de las denominadas cadenas de bloques o Blockchain transformando la forma en que se intercambia valor, debido a que inicialmente blockchain se aplicó en las transacciones con un criptoactivo conocido como Bitcoin.

Sin embargo, estas técnicas han evolucionado a campos que van más allá de la moneda, las finanzas y los mercados, es vista como una innovación técnica y económica, particularmente en ámbitos de servicios, industria y logística; debido a que blockchain cuenta con la capacidad de gestionar transacciones bajo un esquema descentralizado sin la necesidad de entidades centrales que garanticen el cumplimiento de las transacciones o proporcionen confianza en el sistema a los usuarios.

Ahora bien, algunas plataformas que funcionan bajo estructura Blockchain se han diseñado para incorporar los contratos inteligentes (Smart Contracts), como mecanismo que permite eliminar intermediarios para simplificar procesos debido a que es autoejecutable una vez se cumplan las reglas establecidas previamente en la blockchain.

Regresando a la tecnología blockchain, y llevándola al campo del gobierno (denominado también Estado o sector público), potencialmente permitiría a individuos y a comunidades, rediseñar sus interacciones en la política, los servicios y la sociedad en general, con un proceso de desintermediación a gran escala, basada en transacciones automatizadas y en la responsabilidad y seguridad en el manejo de registros oficiales, lo que podría eventualmente, obstruir la corrupción y hacer que los gobiernos sean más eficientes en su misión.

Por lo anterior, integrar el blockchain en el sector público, supondría la solución de muchos de los problemas que la sociedad reclama frecuentemente, sobre todo en temas asociados a la prevención de la corrupción debido a que blockchain automatiza, audita y hace que una aplicación sea transparente pero segura, lo que a la vez podría incurrir en ahorro de costos de infraestructura, dada la relativa simplicidad en las transacciones utilizando smartcontracts.

En síntesis, en esta tesis, se presenta una solución a través de un sistema que por medio de Blockchain público y smartcontracts se presente información de la contratación pública de forma que no se pueda alterar y se mejoren los índices de transparencia de la administración pública colombiana en favor de la disminución de la corrupción. No obstante, para acercar el desarrollo de estas soluciones a las personas no expertas en programación, se presentará un Lenguaje de Dominio Específico -DSL creado mediante la Ingeniería Dirigida por Modelos -MDE que permitirá generar este tipo de aplicaciones para diferentes plataformas y sin requerir conocimientos avanzados de programación y que sean interoperables, escalables y reutilizables.

RESUMEN

Por tanto, a lo largo de esta memoria de tesis se presentará la investigación realizada en el campo de blockchain y de smartcontracts, y concretamente en su aporte en áreas relacionadas con el sector público/estado/gobierno, con el propósito de brindar transparencia en su gestión.

Palabras Clave: Cadena de bloques; contratos inteligentes; redes de pares, Ingeniería Dirigida por Modelos; Lenguajes de Dominio Específico; transparencia, corrupción.

ABSTRACT

The last decade has been characterized by an important development of the so-called blockchains or Blockchain transforming the way in which value is exchanged, due to the fact that initially blockchain was applied in transactions with a cryptoactive known as Bitcoin.

However, these techniques have evolved to fields that go beyond crypto assets, finance and markets, it is seen as a technical and economic innovation, in the areas of services, industry and logistics; because the blockchain has the ability to manage transactions under a decentralized scheme without the need for central entities that guarantee the fulfillment of transactions or provide trust in the system to users.

Now, some platforms that work under the blockchain structure have been designed to incorporate smart contracts, as a mechanism that allows eliminating intermediaries to simplify processes because it is self-executing once the rules previously established in the blockchain.

Returning to blockchain technology, and taking it to the field of government (also called the State or public sector), would potentially allow individuals and communities to redesign their interactions in politics, services and society in general, with a process of disintermediation at large. scale, based on automated transactions and responsibility and security in the management of official records, which could eventually obstruct corruption and make governments more efficient in their mission.

Therefore, integrating the blockchain in the public sector, would mean the solution of many of the problems that society frequently demands, especially on issues associated with the prevention of corruption because blockchain automates, audits and makes an application transparent but secure, which at the same time could incur savings in infrastructure costs, given the relative simplicity in transactions using smartcontracts.

In summary, in this thesis, a solution is presented through a system that, through public blockchain and smartcontracts, presents information on public procurement so that it cannot be altered and the transparency indices of the Colombian public administration are improved. in favor of reducing corruption. However, to bring the development of these solutions closer to non-programming experts, a Specific Domain Language -DSL created through Model-Driven Engineering -MDE will be presented that will allow generating this type of applications for different platforms and without requiring advanced knowledge of programming and that are interoperable, scalable and reusable.

Therefore, throughout this thesis report, the research carried out in the field of blockchain and smartcontracts will be presented, and specifically in its contribution in areas related to the public sector/state/government, with the purpose of providing

transparency in its management.

Keywords: Blockchain; Smart contracts; peer-to-peer networks (P2P), Model-Driven Engineering (MDE); Domain Specific Language; transparency; corruption.

ÍNDICE GENERAL

ÍNDICE DETALLADO

ÍNDICE DETALLADO

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

ÍNDICE DE ANEXOS

BLOQUE I
Planteamiento del problema

ÍNDICE DEL BLOQUE

CAPÍTULO 1. INTRODUCCIÓN

En este primer capítulo se dará una introducción a esta tesis doctoral, presentando las bases de la investigación realizada, iniciando con el problema al que se pretende dar solución, continuando con la motivación para su realización, pasando por las preguntas que dan origen a las hipótesis de esta investigación y los objetivos que se deben alcanzar para demostrar o refutar las hipótesis planteadas.

1. Descripción del problema

La última década, se ha caracterizado por un desarrollo importante del Internet de las Cosas (IoT), por sus siglas en inglés, definida como la interconexión de objetos conectados a Internet con la capacidad de comunicarse (Fang et al., 2016) y que empieza a ser parte de las capacidades tecnológicas de las organizaciones y les permiten lograr niveles de productividad e innovación cada vez más acelerados (Antonio et al., 2018), debido a su objetivo de expandir las comunicaciones existentes persona-persona hacia comunicaciones persona-cosa y cosa-cosa conectando el mundo físico y el mundo virtual (Chung et al., 2013), IoT ha adquirido una gran popularidad debido a la inmensa cantidad de objetos inteligentes existentes que, conectados tienen la capacidad de interactuar entre sí para compartir información y reaccionar según la información que reciben. Actualmente existen plataformas que permiten interconectar objetos de manera centralizada a través de un servidor que se encarga de orquestar todos los objetos inteligentes(Garcia et al., 2014).

A IoT, se suma la aparición de las denominadas cadenas de bloques o Blockchain con la promesa de transformar de manera radical la forma en que se intercambia valor. El Blockchain surgió en 2008, como una propuesta realizada por Satoshi Nakamoto (Nakamoto, 2008), justamente en un momento en el que el deterioro de la economía, la desconfianza crediticia y la crisis hipotecaria, consumía el mercado Estadounidense, y que luego se extendería a otros países (Vigna & Casey, 2015).

La propuesta de Nakamoto pretende reemplazar el modelo centralizado por uno descentralizado, donde el poder de decisión sobre el sistema se delega directamente en los usuarios de la cadena de bloques y se materializó inicialmente en un protocolo denominado Bitcoin que implementaba las reglas de operación de un sistema de gestión de efectivo digital de manera descentralizada sin la necesidad de nodos centrales que lo controlaran o lo regularan (Antonio et al., 2018).

Sin embargo, las técnicas basadas en Blockchain se han independizado y evolucionado del Bitcoin, siendo aplicables a muchos campos que van más allá de la moneda, las finanzas y los mercados, es vista como una innovación técnica y económica(Cong & He, 2018), (Macrinici et al., 2018), (Savelyev, 2017) particularmente en las áreas de gobierno, salud, ciencia, alfabetización, cultura y arte (Nathan & Scobell, 2012), (Portmann, 2018). La tecnología Blockchain se identifica como factor clave en la resolución de problemas de escalabilidad, privacidad y confiabilidad relacionados directamente con el paradigma de IoT (Reyna et al., 2018)

El principal potencial revolucionario y transformador de la tecnología Blockchain se encuentra en la capacidad de gestionar activos digitales y facilitar su transferencia bajo un esquema descentralizado (como se enunció anteriormente) (Ali et al., 2018) , sin la necesidad de entidades centrales que garanticen el cumplimiento de las transacciones o proporcionen confianza en el sistema a los usuarios (Preukschat Carlos Kuchkovsky et al., 2017). A su vez, se destaca como principal ventaja del Blockchain, que automatiza, audita y hace que una aplicación sea transparente pero segura; puede prevenir fraude y corrupción y se puede utilizar para resolver muchos otros problemas dependiendo de cómo se implemente (Prusty, 2017), (Viriyasitavat & Hoonsopon, 2019).

Lo anterior, debido a que Blockchain es una estructura de datos en forma de lista de bloques encadenados, cada uno de ellos contiene un enlace encriptado al bloque anterior por medio de un algoritmo de transformación de claves (hash) y se encuentra distribuida en redes de pares (red peer-to-peer), que permite verificar la integridad de la información a través de todos los participantes de la red de pares, sin la necesidad de una autoridad confiable (Christidis & Devetsikiotis, 2016).

Las técnicas de Blockchain se utilizan en Internet de las cosas para asignar identificadores únicos a los distintos objetos inteligentes, de forma que se incremente la seguridad respecto a posibles ataques de suplantación de identidad (Manuel et al., 2019), debido a que se conservan atributos de IoT, entre los cuales se encuentran el almacenamiento seguro, es casi imposible de falsificar y se puede utilizar para muchos aspectos en el negocio electrónico(Cong & He, 2018)

La estructura Blockchain se hace cargo de transacciones, acuerdos, registros de propiedad e innovaciones, desarrollos y otros bienes; de igual forma se diseña para soportar los contratos inteligentes (Smart Contracts), que son mecanismos que tienen como objetivo eliminar intermediarios para simplificar procesos.

Un contrato inteligente es capaz de ejecutarse y hacerse cumplir por sí mismo, de manera autónoma y automática, sin intermediarios, debido a que en el contexto Blockchain, son scripts (códigos informáticos) almacenados en Blockchain, como residen en la cadena, tienen una dirección única y se activan al realizarle una transacción, acto seguido, se ejecuta de forma independiente y automática en cada nodo de la red (de forma predeterminada), en concordancia con los datos que se incluyeron en la transacción desencadenante (Reyna et al., 2018),(Christidis & Devetsikiotis, 2016). Por lo anterior, un contrato inteligente tiene validez, sin depender de autoridades o de terceros (Cong & He, 2018) sino por consenso de los usuarios de la red (Giancaspro, 2017),(Shermin, 2017); eliminando la burocracia dado el carácter descentralizado, inmutable y transparente de la tecnología Blockchain.

Regresando a la tecnología Blockchain, y llevándola al campo del gobierno/Estado, se encuentra que potencialmente permite a individuos y comunidades, rediseñar sus interacciones en la política, los negocios y la sociedad en general, con un proceso de desintermediación a gran escala, basada en transacciones automatizadas y en la responsabilidad y seguridad en el manejo de registros oficiales (Reijers et al., 2016), (Hou, 2017), lo que podría eventualmente obstruir la corrupción y hacer que los servicios gubernamentales sean más eficientes (Casino et al., 2019), tornándose relevante para la adopción de políticas públicas y de desatar procesos de innovación

social que requieren los países de américa latina para el logro de los retos sociales y económicos presentes y futuros.

Por lo anterior, el Blockchain en la gestión pública podría constituir soluciones descentralizadas y como repositorios públicos impulsados por consenso, que pueden tener una serie de aplicaciones para hacer ciudadanos menos dependientes de los gobiernos, pero dentro de una sociedad que en última instancia se basa en la autoridad del Estado. Así las cosas, proporcionar los mismos servicios que ofrece el estado y las autoridades públicas correspondientes (manteniendo su validez), de manera descentralizada y eficiente a través de Blockchain no significa despedir al Estado, sino promover el buen gobierno; esto es "hacer mejores gobiernos cuando no se concentra el todo el poder en manos de unas pocas personas" (Casino et al., 2019),(Andreas Antonopoulos, 2016).

Según Preukschat(Preukschat Carlos Kuchkovsky et al., 2017), la administración pública está en plena crisis ya que el modo de entenderla está cambiando, para adaptarse a estos cambios, es necesario impulsar nuevas medidas que garanticen un modelo activo e inteligente, dado que la ciudadanía exige una administración más transparente, rápida y eficiente.

Por ello, integrar el Blockchain en la gestión pública, supondría la solución de muchos de los problemas que la sociedad reclama frecuentemente, e influyendo directamente a las organizaciones, a la vez que se podría incurrir en ahorro de costos de infraestructura, dada la relativa simplicidad en las transacciones utilizando los contratos inteligentes(Giancaspro, 2017).

Finalmente, es necesario enunciar la Ingeniería dirigida por modelos (Model-Driven Engineering, MDE), que juega un papel imprescindible en la investigación y desarrollo de nuevas tecnologías, por cuanto pretende dar solución al problema de las plataformas cambiantes en las empresas que asumen el uso de sistemas informáticos, la interoperabilidad y la portabilidad de los mismos (Martínez et al., 2015) y su uso permite reducir la complejidad del desarrollo de software automatizando tareas y procesos, logrando así software más fiable y con funcionalidades más sofisticadas (Vicente García-Díaz et al., 2010),(B. Selic., 2008).

Adicionalmente, la ingeniería dirigida por modelos (MDE, Model-Driven Engineering) ofrece garantías de reusabilidad de código e interoperabilidad entre aplicaciones, lo que implica el poder reutilizar ciertas partes o ciertos lenguajes de dominio específico para crear un desarrollo confiable, de calidad y automatizable (B. Selic., 2008).

Para el caso de estudio, la IoT y la tecnología Blockchain requieren reconocer los objetos inteligentes, mantener un flujo de mensajes y de instrucciones entre los diferentes objetos. Sin embargo, cada implementación puede presentar diferentes problemas y cada aplicación proporciona una solución diferente.

Una solución para estos casos es realizar una arquitectura que soporte el paso de mensajes de los diferentes tipos de dispositivo y sea capaz de responderles (Chung et al., 2013), (Gama et al., 2012), a partir de aproximaciones mediante MDE, con la cual se propone la creación de aplicaciones que permitan interconectar objetos heterogéneos (García & Espada, 2013), siendo estas la base en el diseño de un meta-

modelo de contratos inteligentes usando Blockchain aplicado al sector público.

En esta tesis, se presenta un sistema fiable que por medio de Blockchain público, muestre los Smart Contracts de forma que no se pueda alterar y se mejoren los índices de transparencia de la administración pública colombiana en favor de la disminución de la corrupción.

Este sistema puede ser observado por las entidades públicas para el diseño, desarrollo y operación de procesos de contractuales, basados en la tecnología blockchain y Smart contracts, de forma organizada, escalonada y estructurada, permitiendo de manera adicional a lo enunciado anteriormente, presentar iniciativas que encajan sin fricciones dentro de la ya generalizada y aceptada política de gobierno abierto promovida por la Comisión Económica para América Latina y el Caribe (CEPAL), cuyos preceptos son los de transparencia, colaboración, participación y datos públicos abiertos u "Open Government Data"(CEPAL-UN, n.d.).

2. Hipótesis

Internet de las Cosas aunado a la aparición del blockchain se constituye en un campo tecnológico en auge con gran potencial para la resolución de problemas en diversos ámbitos, como se ha enunciado en el apartado anterior. Ante el estudio de estos problemas específicamente en áreas de gobierno, ha surgido una pregunta de investigación a partir de la cual se formula la hipótesis que esta tesis doctoral pretende contrastar.

¿Es posible la integración de blockchain y los smart contracts a través de un meta-modelo aplicado al sector público?

Teniendo en cuenta la pregunta de investigación, surge la hipótesis de esta tesis doctoral y que se deriva en las soluciones específicas abordadas:

> La integración de Internet de las cosas y el Blockchain, permite reconocer la importancia de los contratos inteligentes en el desarrollo de aplicaciones en áreas asociadas al Estado (también denominado gobierno o sector público) en su lucha contra la corrupción, debido a su versatilidad, seguridad, acceso y control del trámite o proceso que se esté realizando; dado que se cuenta con un sistema de información permanente y público.

3. Objetivos

En este apartado se presentan los objetivos derivados a partir de las preguntas de investigación planteadas en el apartado anterior y que originaron la hipótesis de esta tesis doctoral. Estos objetivos se deben cumplir para comprobar la hipótesis planteada. Primero se presentará el objetivo principal y a continuación los objetivos específicos en los que se divide el objetivo principal.

El objetivo principal de esta tesis doctoral es el siguiente:

Diseñar un sistema fiable que por medio de Blockchain público, muestre los Smart Contracts de forma que no se pueda alterar y se mejoren los índices de transparencia de la administración pública colombiana en favor de la disminución de la corrupción.

Este objetivo principal se divide en los siguientes objetivos específicos, que se abordan en cada solución presentada:

- Elaborar el estado del arte de Internet de las Cosas, Blockchain y los Smart Contracts.
- Especificar las herramientas tecnológicas aplicables y los requerimientos para la propuesta de un sistema o meta-modelo de integración Blockchain y Smart Contracts.
- Analizar, diseñar, desarrollar e implementar un prototipo de sistema, modelos y meta-modelo de integración Blockchain y Smart Contracts.
- Proponer y aplicar pruebas de validación para la propuesta.

CAPÍTULO 2. EL PROCESO DE INVESTIGACIÓN

Una vez de introduce la tesis doctoral, en este segundo capítulo se presenta la metodología que se ha usado para su desarrollo. El primer apartado aborda de manera detallada la metodología de trabajo, las líneas de investigación y los métodos científicos utilizados. Por último, se presenta la organización de esta memoria de tesis doctoral, estructurada tanto en bloques como en capítulos para ofrecer al lector una mejor visión general del contenido.

1. Metodología de trabajo

El trabajo de investigación realizado en esta tesis doctoral ha sido desarrollado en varias fases utilizando un enfoque incremental. En la Figura 1, se muestra el marco metodológico de la investigación con referencias temporales, así como el desarrollo temporal de la investigación a gran escala.

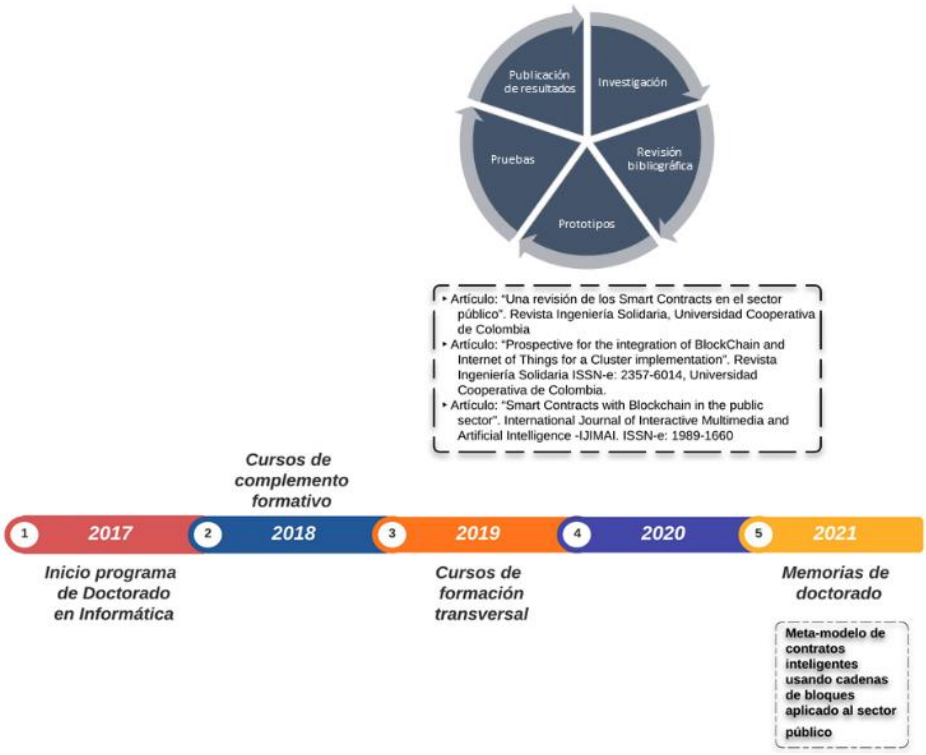


Figura 1. Metodología utilizada y desarrollo temporal a gran escala de esta tesis doctoral

El inicio del proceso en el programa de Doctorado en Informática en la Universidad de Oviedo data de 2017 cuando se accede al mismo, posteriormente se cursan los complementos formativos de *Diseño y Construcción de MDA* y *Lenguajes y Plataformas de Programación,* para cimentar los conocimientos necesarios para el desarrollo de la investigación conducente a esta tesis doctoral. Por otro lado, se re realizaron los cursos de formación transversal como requisito para avanzar en el programa.

En concomitancia, se da la investigación realizada, cuyo proceso y resultado se presenta en esta tesis doctoral y se enmarca en la línea de investigación de ***Ingeniería del software***

y gestión de la información.

Así las cosas, se parte del estudio del estado del arte del block chain y Smart contracts, desde una exhaustiva revisión biliográfica que ha servido como base sólida para la realización de la investigación aquí presentada.

Para llevar a cabo esta tesis doctoral se ha optado por una **metodología de desarrollo ágil basado en el método científico** y **ejecutando en paralelo diferentes investigaciones** enmarcadas dentro de la línea de investigación de Ingeniería del software y gestión de la información. Los resultados parciales obtenidos de la investigación realizada, se han enviado y han sido publicados en revistas.

En la parte superior derecha de la Figura 1 se observan las fases principales abordadas para el desarrollo esta tesis, a saber:

1. **Investigación**: En esta fase se realiza una investigación del trabajo previo acerca de block chain y Smart contracts, recopilando información de distintas fuentes de publicaciones científicas y como resultado de esta revisión bibliográfica se formulan los objetivos y la hipótesis de la investigación. En esta fase, también se identifican y evaluan aplicaciones y plataformas de block chain y Smart contracts en el sector público y en el sector público en Colombia, estableciendo sus interacciones.
2. **Prototipos**: En esta fase se seleccionan las herramientas y tecnologías aplicables, se identifican los requerimientos para proproner el sistema o meta-modelo de integración entre block chain y Smart contracts, se modeliza, analiza, propone y desarrolla el prototipo o los prototipos que pretenden validar la hipótesis planteada y cumplir con los objetivos.
3. **Pruebas**: En esta fase se realiza la comprobación de prototipos, se estudia su rendimiento, se aplican las diferentes pruebas de validación y evaluación y se refina el prototipo con el propósito de validar la hipótesis y planteamientos.
4. **Publicación de resultados**: De manera simultanea con las demás fases, los resultados de la investigación se documentan y se difunden en la comunicada científica mediante publicaciones materializadas en artículos para ser publicados en revistas. Adicionalmente, se documenta esta tesis doctoral.

Durante este proceso se realizó la publicación de cinco (5) artículos, uno de ellos en revista perteneciente al primer cuartil del índice JCR.

Hacia finales de 2017 fui admitida al Doctorado en Informática, con dedicación de tiempo completo, no obstante, a mediados de 2018 se presentó el cambio de la dedicación de la investigación a tiempo parcial.

2. Desarrollo temporal de la investigación

En este apartado se presentan los hitos y eventos más significativos surgidos en el desarrollo de esta tesis doctoral y que se muestran a gran escala en la Figura 1:

- **Octubre de 2017**: Se da inicio al Doctorado en Informática de la Universidad de Oviedo, centrando la temática de investigación en block chain y Smart contracts, dando origen al título de esta tesis: «*Meta-modelo de contratos inteligentes*

usando cadenas de bloques aplicado al sector público».

- **Septiembre de 2018**: Se cursa y aprueba el primer curso de complemento formativo del doctorado. El título del curso es «*Diseño y Construcción de MDA*». En este mismo periodo, se publica el artículo: «Modeling and Simulation of Integration of Internet of Things and Manufacturing Industry 4.0».

- **Noviembre de 2018**: Se cursa y aprueba el segundo curso de complemento formativo del doctorado. El título del curso es «*Lenguajes y Plataformas de Programación*».

- **Febrero a Julio de 2019**: Se cursan y aprueban los (3) tres cursos de formación transversal del doctorado, que se constituyen como requisito para adelantar el programa de Doctorado en Informática. Los respectivos títulos de los cursos son: **1.** «*Iniciación a la Edición y Procesamiento de textos en Latex. Primera parte. Utilización de plantillas y Edición básica*»; **2.** *«Iniciación a la edición y procesamiento de textos en Latex. Parte 2. Entorno gráfico, presentaciones y plantillas»* y; **3***. «Edición digital académica de fuentes documentales para los estudios históricos».*

- **Septiembre de 2020**: **Publicación** de dos (2) artículos como resultado del avance en la investigación, titulados: **1.** *«Prospective for the integration of BlockChain and Internet of Things for a Cluster implementation*» *y* **2.** *«Smart Contracts with Blockchain in the public sector*»

- **Diciembre de 2020**: **Publicación** de dos (2) artículos como resultado de los cursos de complemento formativo, titulados: **1.** *«MEVF "Fiscal Surveillance Entities Model" a DSL proposed for interoperability and management of fiscal scenarios*» *y;* **2.** *«Run-Time optimization using the invokedynamic statement*»

- **Junio de 2021**: Conclusión y elaboración de esta memoria de tesis.

3. Organización de la memoria de tesis

Esta memoria de tesis doctoral está estructurada en bloques que agrupan los capítulos que están relacionados. En este apartado se presenta de forma resumida la organización de esta tesis para facilitar su lectura.

- **Bloque I – Planteamiento del problema**: Este bloque está conformado por dos (2) capítulos introductorios. En el primero de ellos, el Capítulo 1 titulado «Introducción» se presentan la motivación para realizar esta tesis describiendo el problema que se pretende resolver, la hipótesis y los objetivos planteados. Por su parte, en el Capítulo 2, titulado «El proceso de investigación» se presenta la metodología de trabajo que se siguió, el desarrollo temporal y la organización de esta memoria para facilitar la ubicación del lector.

- **Bloque II – Marco Teórico**: Este bloque contiene los capítulos que abordan el estado del arte y el trabajo relacionado con lo estudiado en esta tesis doctoral. Los contenidos que se abordan en este bloque son los siguientes: BlockChain (Capítulo 3), Smart Contracts (Capítulo 4), Aplicaciones de block chain y smart contracts en

el sector público (Capítulo 5), Potencial de block chain en la contratación pública (Capítulo 6) e ingeniería basada en modelos (Capítulo 7).

- **Bloque III – Solución propuesta y meta-modelo**: Este bloque contiene la solución y el meta-modelo propuesto el problema planteado en esta tesis doctoral, presentado en el capítulo 8, la validación y las pruebas en el capítulo 9, y los resultados en el capítulo 10.

- **Bloque IV – Conclusiones y trabajo futuro**: Este bloque se encuentra constituido por cuatro (4) capítulos. El Capítulo 11 presenta las conclusiones de esta tesis doctoral; el Capítulo 12, por su parte presenta el trabajo futuro que se puede realizar a partir de esta tesis. En el Capítulo 13 se presentan las publicaciones derivadas de esta tesis doctoral y finalmente, en el Capítulo 14 se encuentra la Bibliografía que contiene todas las referencias utilizadas en esta tesis doctoral.

BLOQUE II
Marco Teórico

ÍNDICE DEL BLOQUE

CAPÍTULO 3. BLOCKCHAIN

En este capítulo se introducirá una de las tecnologías más importante de esta tesis doctoral, **blockchain**. Se enunciará su significado y cómo surgió esta tecnología, se hará un repaso sobre sus distintas arquitecturas, consensos para validación de datos, permisos e integración en blockchain, así como se hará una breve presentación de los diferentes campos de aplicación de esta tecnología.

1. Definición y antecedentes

El **Blockchain**, o ***cadena de bloques*** surgió en 2008, como una propuesta realizada por Satoshi Nakamoto (Nakamoto, 2008), en un momento en el que el deterioro de la economía, la desconfianza crediticia y la crisis hipotecaria, consumía el mercado Estadounidense, y que luego se extendería a otros países (Vigna & Casey, 2015).

Este término dio popularidad no solamente al bitcoin sino que también a él se debe el éxito de otras criptomonedas (Ardila, 2018), de hecho, con el paso del tiempo debido a las ventajas y propiedadades de la implementación de blockchain, se ha identificado su gran potencial para ser aplicada en otros ámbitos, que serán presentados en el apartado 7.

Su origen se encuentra relacionado con la criptografía, ligada a las guerras y a las luchas de poder entre los estados, cuyo uso se intensificó en los años noventa con el criptoanarquismo, enmarcado en el movimiento "cypherpunks" o de activistas que se oponen a la vigilancia de las redes informáticas por parte de los estados y evaden la censura, defienden la generalización de la criptografía y las tecnologías que mejoran la privacidad (Preukschat Carlos Kuchkovsky et al., 2017).

Se trata entonces el Blockchain, de una tecnología o sistema para gestión y almacenamiento de datos de manera descentralizada, autónoma, auditable y confiable (García Mateo, 2018); es una estructura de datos distribuida utilizada para crear un registro público, validado y autorizado de todas las transacciones que los usuarios de la red hayan ejecutado (Nakamoto, 2008), (Smith, 2011), (Chung et al., 2013), conocido como ***ledger,*** en forma de lista de bloques encadenados en una forma secuencial creciente, que toma información de manera segura en redes **P2P o peer-to-peer** (de igual a igual) en la que los miembros que no confían pueden interactuar entre sí, de manera verificable sin un intermediario de confianza (Christidis & Devetsikiotis, 2016), que se replica y comparte entre los miembros de la red.

Expuesto lo anterior, es importante recordar que una blockchain no es una base de datos tradicional, debido a que el control de acceso y lectura de datos de esta última está centralizado. En el caso de blockchain, estos controles se encuentran verdaderamente descentralizados, adicionalmente, se diferencia debido a la capacidad que tiene blockchain para asegurar transacciones sin necesidad de terceros de confianza en un entorno competitivo (Drescher, 2018), como se evidencia en la figura:

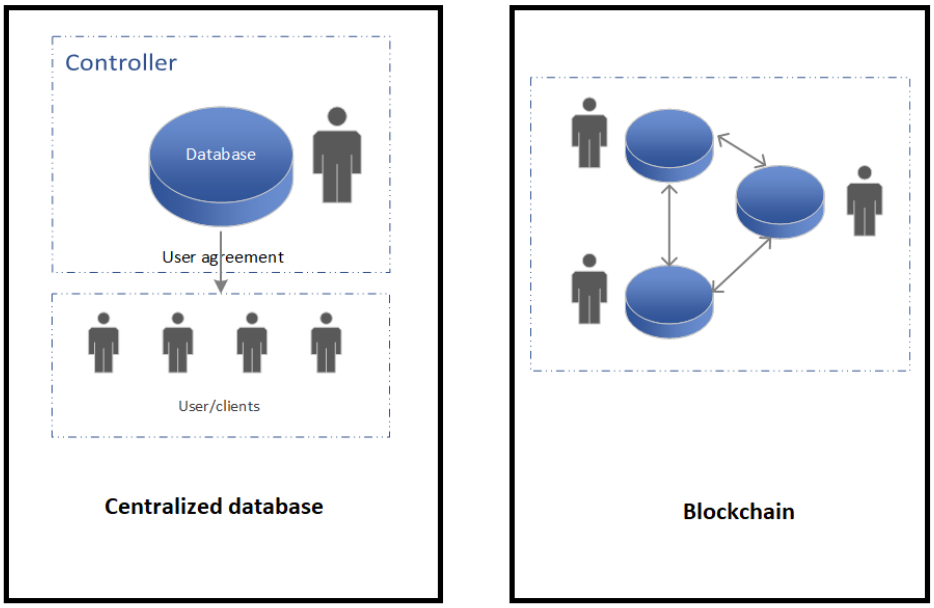


Figura 2. Representación de una base de datos centralizada vs blockchain Basado en (Singh, 2017)

2. Funcionamiento del Blockchain

La tecnología blockchain se encuentra conformada por un *nodo*, un *protocolo estándar*, una *red de pares* y un *sistema descentralizado*. A continuación, se presentará cada uno brevemente:

- **Nodo:** Puede ser un ordenador personal o una megacomputadora, según la complejidad de la red. No obstante, todos los nodos deben poseer el mismo protocolo para comunicarse entre sí para poder conectarse y formar parte de la red de una blockchain, independientemente de su tipología (pública, privada, híbrida).

- **Protocolo**: Otorgan un estándar común para definir la comunicación entre los ordenadores participantes en la red, se presenta en forma de software informático para que una red de nodos pueda comunicarse entre sí.

- **Red entre pares o peer-to-peer (P2P)**: Red de nodos conectados directamente en una misma red, en la que la información almacenada funciona sin necesidad de un cliente o servidor fijo.

Una aplicación que se base en una red P2P para transmitir información entre los nodos debe tener en cuenta que cada uno de ellos cuenta con la instancia de la aplicación ejecutándose y escuchando a la red. Los nuevos nodos en la red se dan de alta y por ello debe existir uno o más nodos conocidos a priori, para solicitar su acceso o un sistema para dar de alta nodos a la red.

Las redes P2P se caracterizan porque: se incrementa su *robustez*, ya que cuenta con la réplica de la información en múltiples destinos y en sistemas P2P puros no hay ningún punto crítico en donde pueda haber un fallo y afecte a los demás nodos de la red; son

descentralizadas; son *escalables*, toda vez que las redes P2P tienen un alcance variable y lo deseable es que mientras más nodos estén conectados a esta red, mejor sea su funcionamiento y con ello aumenten los recursos totales del sistema; y la *seguridad*, en el sentido en el que a estas redes se incorporan diversas funciones tendientes a controlar los nodos de la red que sean maliciosos.

En la figura que se presenta a continuación, se muestra la tipología de una red P2P y una red tradicional de servidor-cliente:

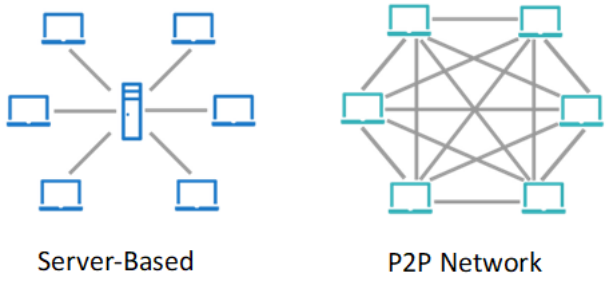


Figura 3. Representación de la tipología de usuario-servidor y P2P (Suárez Alvaro, n.d.)

Como lo presenta la figura 3, en una red tradicional Cliente-Servidor cuenta con la información centralizada en uno o varios puntos fijos, mientras en una red P2P la información se transmite hacia todos los nodos de la red y no hacia puntos fijos.

- **Sistema descentralizado:** La información está controlada por todos los nodos u ordenadores porque todos son iguales entre sí, es decir que no existe jerarquía entre los nodos, al menos en una blockchain pública.

Una vez descritos los elementos que conforman una blockchain, se pasará a describir su funcionamiento:

El primer bloque de la blockchain o el bloque jamás creado es conocido como el ***bloque génesis*** (no tiene bloque predecesor y es el único bloque que difiere de los demás), es el elemento encargado de iniciar el sistema para permitir la comunicación y persistencia de todos los siguientes bloques siendo como el bloque fuente de todos los siguientes. Por ejemplo, en la blockchain de bitcoin este bloque fue minado el 3 de enero de 2009 y uno de los campos minados contenía un titular del diario "*The Times*" de esa fecha, con el fin de manifestar la prueba que ese bloque había sido creado ese día.

Ahora bien, a la blockchain se agregan bloques que contienen transacciones, y con el apoyo de la función hash y de la clave ***hash*** (resultado de una función criptográfica *hash* que se trata de un algoritmo matemático para transformar un conjunto de datos sin atender a su estructura lingüística, en una serie de caracteres con una longitud fija que sea difícilmente comprensible).

Cada nuevo bloque generado o transacción, cuenta con su propio *hash* y a partir de un número aleatorio llamado nonce (que se debe adivinar), se combina con los datos contenidos en el bloque para crear a partir de la función hash, la clave hash del nuevo bloque candidato o transacción para posteriormente verificarlo y validarlo, ejecutando algortimos de consenso. El proceso se trata de un cálculo computacional considerable, ya

que de debe adivinar el nonce, siendo la dificultad mayor a medida que la blockchain es más grande.

La creación de nuevos bloques es realizada por nodos ***mineros*** o nodos de la red que participan en el proceso de escritura de datos en la blockchain a cambio de una recompensa (C. R. Dolader et al., 2017). Ahora bien, esta información registrada en la red **P2P** es revisada, validada y aceptada por el resto de los participantes mediante un consenso que permite establecer garantías entre los mineros de la blockchain (de los consensos se tratará en el apartado 3).

El bloque validado se une a la cadena que está secuenciada por eventos previos que contienen su *hash* inmodificable que lo identifica. Como cada bloque contiene un *hash del bloque anterior*, un *hash propio*, una marca de tiempo y los datos de transacción; se va creando una cadena de bloques vinculados entre sí, por lo que la blockchain contendrá información completa e íntegra sobre los balances del bloque inicial y al tener cada uno de ellos un enlace encriptado al bloque anterior por medio del algoritmo de hash, la información de cada bloque solamente puede ser cambiada modificando todos los bloques posteriores en la lista(Cueva-Lovelle, 2019).

En resumen, se puede afirmar que el funcionamiento del blockchain está dado por un conjunto de servidores u ordenadores denominados “nodos”, conectados en red utilizan un mismo sistema de comunicación (el protocolo) con el objetivo de validar y almacenar la misma información registrada en la red P2P, la información recogida no puede modificarse porque complejos algoritmos criptográficos, sumados a la propia capacidad colectiva de la red, contribuyen a asegurar la irreversibilidad de la información (Preukschat Carlos Kuchkovsky et al., 2017).

Para finalizar la descripción del funcionamiento de la blockchain, en la figura se muestra la información que contiene cada bloque en una blockchain genérica, que da claridad frente a lo planteado en las líneas anteriores:

- ***Hash del bloque actual***: permite realizar consultas acerca de la información contenida en el bloque, de forma eficiente y segura.
- ***Datos*****:** Es la información en sí del bloque.
- ***Timestamp*****:** Es la marca de tiempo que permite identificar el instante en el que fue creado el bloque.
- ***Nonce*****:** Number used once. Consiste en un número aleatorio que es útil en la búsqueda de una función hash que dé respuesta al problema matemático propuesto a la hora de minar un bloque (proceso competitivo que sirve para validar un nuevo bloque). Es el valor encontrado por potencia de cálculo en el proceso de minado.
- ***Hash del bloque previo***. Este valor permite que los bloques queden vinculados secuencialmente formando una cadena.

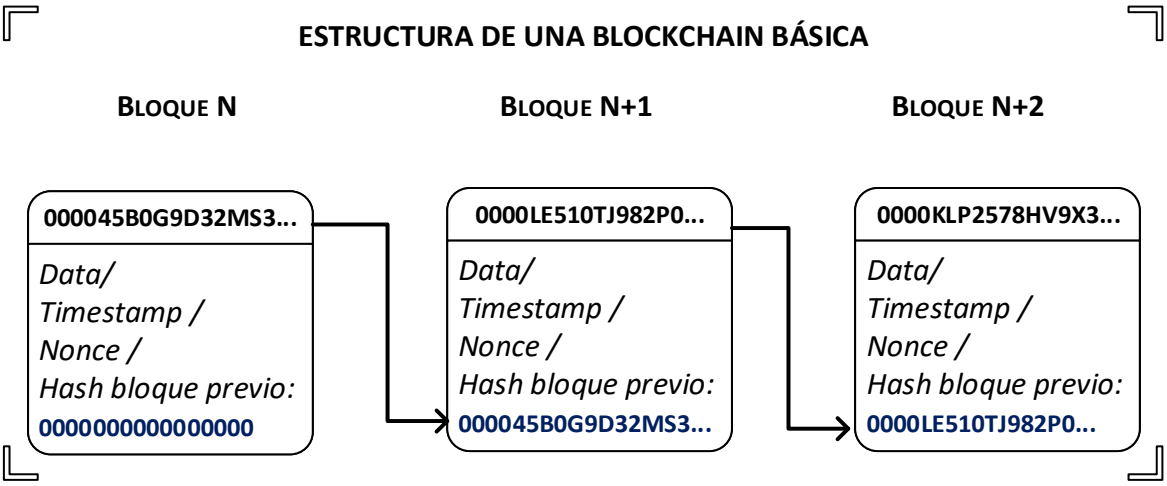


Figura 4. Representación gráfica de la estructura de una blockchain básica. Elaboración propia

3. Consensos para la validación de datos y permisos en Blockchain.

La validación de la información se lleva a cabo mediante un mecanismo llamado ***consenso***, es básicamente un concepto de computación distribuida que se ha utilizado en blockchain para proporcionar un medio para aceptar una única versión de la verdad por parte de todos los pares en la red blockchain(Bashir, 2017a)

Los mecanismos de consenso son algoritmos matemáticos distribuidos que están en el núcleo del funcionamiento de las tecnologías de registro distribuido DLT (del inglés, "Distributed Ledger Technology") y gestionan los acuerdos entre los nodos de la red para llegar a una única verdad.

Este mecanismo permite validar una decisión (nuevo bloque o transacción, para el caso de blockchain) entre todos los nodos y es el que define si un registro o información se puede inscribir o no, en un bloque.

La principal dificultad que se enfrentan los algoritmos de consenso es la tolerancia frente a fallos, ya que en los sistemas distribuidos son propensos a fallos bien por fallos intencionados (problema de los generales bizantinos) o por caídas. Su responsabilidad se basa en mantener la consistencia entre los nodos que comparten la información en una red P2P.

Una red blockchain puede "ponerse de acuerdo" en una transacción de muchas formas, dependiendo del ámbito de aplicación del proceso que esté sucediendo, a través de *protocolos de consenso*.

Es importante destacar que cada protocolo cuenta con características diversas que sirven mejor para unos propósitos y unos tipos de blockchain, que para otros.

En la actualidad se observan distintas implementaciones de protocolos de consenso y su clasificación difiere, por ejemplo, Bashir (Bashir, 2017a) clasifica los mecanismos de consenso en dos categorías:

- Basado en pruebas (Proof-based), basado en líderes (leader-based), o el consenso de Nakamoto por el cual se elige a un líder y propone un valor final.

- Basado en la tolerancia a fallas bizantina, que es un enfoque más tradicional basado en rondas de votación.

Ocariz(Ocariz Emiliano B., 2019) por su parte, enuncia los siguientes mecanismos de consenso: *Proof of Work, Proof of Authority, Proof of Weight, Proof of Capacity, Proof of Elapsed Time, Proof of Burn, Proof of Stake (prueba de participación), Delegated Proof of Stake (prueba de participación delegada), Proof of Activity*, *Federated Byzantine Agreement* y *Direct Acyclic Graphs (DAG). No* obstante, el mismo autor aclara que este listado no es exhaustivo, debido a que pueden existir otros protocolos no formalizados, presentan barreras o se encuentran aún muy difusos.

A continuación, se desarrollarán los principales protocolos de consenso, es decir: protocolo prueba de trabajo (PoW), protocolo prueba de participación (PoS) y prueba de participación delegada (DPoS).

3.1 Prueba de trabajo PoW (del inglés, proof-of-work)

El protocolo ***Prueba de trabajo (PoW)*** nace para evitar que una mayoría de nodos pudieran mantener comportamientos indeseados o que los nodos registren bloques al azar, así como para establecer un mecanismo de información para que la blockchain sepa qué nodos han conseguido el registro de las transacciones y evitando que estos se puedan alterar.

Este tipo de mecanismo de consenso se basa en la prueba de que se han utilizado suficientes recursos computacionales antes de proponer un valor de aceptación por parte de la red. Es uno de los consensos más populares, especialmente en los criptoactivos.

Actualmente, este es el único algoritmo que ha demostrado ser exitoso contra *Sybil attacks*(Bashir, 2017a) (ataque en el cual un sistema de reputación es subvertido mediante la creación de múltiples identidades en una red P2P).

Cuando reciben una solicitud de transacción, los mineros comprueban que dicha transacción se puede realizar. Para verificar esta información, los mineros acuden a la copia de la cadena de bloques que cada uno de ellos almacena, la cual tiene registrados todos los movimientos desde su génesis. De esta forma, pueden saber con certeza si las operaciones se pueden llevar a cabo o no. Una vez verificados los datos, el minero añadirá a su bloque de transacciones esa operación válida.

Sin embargo, para evitar la difícil tarea de corromper una cadena de bloques, el protocolo PoW hace que los mineros tengan que competir entre ellos para dar con la solución al problema matemático, para la cual se obtiene a partir de operaciones matemáticas que se deben realizar a los bloques.

Son las funciones hash, normalmente SHA-256, donde se irán probando pequeñas variaciones a la entrada del bloque, hasta encontrar una válida que devuelva el valor hash que se está buscando, convirtiéndose en su identificador en la blockchain.

Una vez encontrada esa solución, el bloque se convertirá en una parte de la cadena y esto sucederá cada vez que la mayoría de los mineros lleguen a los consensos de determinación de las transacciones registradas por el minero como válidas y, que se ha adivinado

correctamente el valor "Nonce" logrando un valor hash del bloque con un determinado número de ceros al principio. La comunidad de mineros comprueba estos datos a través de la firma digital (hash) del bloque ganador.

Como cualquier trabajador, el minero también tiene derecho a recibir una compensación o remuneración por su trabajo.

Es importante destacar que debido a las características de la función de hash, no es posible calcular estos valores analíticamente, es decir, para obtener un bloque válido, el minero debe recurrir a la fuerza bruta: probar valores del parámetro nonce hasta hallar uno válido. El proceso de probar valores o fuerza bruta es un proceso computacionalmente costoso, de ahí que este mecanismo se conozca como "prueba de trabajo".

Si se tiene un blockchain público que usa PoW como mecanismo de consenso, el anonimato del usuario es alto y la inmutabilidad de los datos es moderada, pero el inconveniente es que su capacidad de escala es baja.

La ventaja de este protocolo es la seguridad debido a que resulta costoso romperla, sin embargo, su aplicación también implica un importante gasto económico y ambiental en los procesos de minería y la necesidad de disponer de gran cantidad de recursos computacionales.

3.2 Prueba de participación PoS (del inglés, proof-of-stake)

La prueba de participación o proof of stake ***(PoS),*** es una alternativa al *PoW* para blockchain públicas y también se adecua a las blockchain de consorcios que utilizan el consenso de acuerdo bizantino federado.

De la misma forma que el *PoW*, la prueba de participación *PoS* busca conseguir el consenso entre los nodos para validar los bloques, pero con la realización de cálculos más sencillos que no dependen del cálculo intensivo en busca de una recompensa, y teniendo en cuenta la optimización en el gasto de recursos para reducir el coste asociado, teniendo en cuenta que no existen recompensas por los bloques minados como en *PoW*, sino que los mineros se deben ajustar a las tarifas por transacción.

Este protocolo asigna una probabilidad de crear bloques, proporcional a la cantidad de tokens que posee cada nodo. Así, aquellos usuarios que disponen de una participación predominante en la red al haber conseguido con anterioridad un porcentaje mayor de tokens serán los que creen bloques con más frecuencia(Dolader et al., 2017).

En PoS existen nodos validadores del bloque que son seleccionados de acuerdo a su "stake", es decir, a su participación en el sistema. Esta participación está relacionada con su poder económico y la cantidad de tokens que posee, por ello son los más interesados en el buen funcionamiento de la red; y dependiendo de la implementación del PoS el nodo que se postule a nodo validador deberá dejar en una especie de baúl el monto que arriesga, de manera que no podrá utilizar estos tokens en forma inmediata. La premisa es que quien tenga más tokens en juego no atentará contra el sistema porque también estará atentando contra sí mismo.

Entonces, los nodos validadores serán un grupo seleccionado en relación con su

"participación", pero también con una selección aleatoria para evitar que aquellos que más tienen se queden siempre con la recompensa.

Ocariz aclara que, la recompensa en la mayoría de los sistemas de PoS está compuesta por el pago de cargos por transacción cobrados a cada una de las personas que realizaron intercambios y su transacción fue ubicada dentro del bloque validado(Ocariz Emiliano B., 2019).

Adicionalmente, señala como particularidad del PoS que no se habla de proceso de minería, sino de "minting", que consume considerablemente menos recursos energéticos, no requiere potencia de computación para su buen funcionamiento y es más veloz, permitiendo un gran número de validaciones en menos tiempo.

La razón por la cual PoS es más liviano es que no debe resolverse el complejo cálculo donde se debía encontrar un número comenzando por una cantidad determinada de ceros.

Lo anterior podría significar que PoS es más centralizado y que tienen más poder quienes más criptoactivos tienen (Ocariz Emiliano B., 2019), (Dolader et al., 2017). Para evitar que esto pase han surgido diferentes métodos que solventan el problema. En ellos se pondera el stake de cada propietario junto con la duración que hace que este lo posea con tal de asignar una probabilidad para la creación del próximo bloque.

El beneficio más destacado de la aplicación de este mecanismo de consenso es que al no requerir una enorme cantidad de energía para los procesos computacionales, se convierte en una alternativa mas económica y con menor impacto ambiental.

3.3 Prueba de participación delegada DPoS

Este mecanismo de consenso es una versión del ***PoS*** en el que se seleccionan nodos ***testigos*** o ***witnesses***. Estos testigos son elegidos con base en votos que tienen un peso acorde al tamaño de su stake o de su participación, es decir, a la cantidad de tokens que posee. Una vez elegidos, el grupo de testigos valida las transacciones y crea los bloques, y por ello obtiene cierta cantidad de tokens como recompensa, es decir que este grupo hace la tarea equivalente a la efectuada por los mineros en *PoW.*

El incentivo económico evita que los nodos *testigos* actuen en contra del sistema y se genera una competencia permanente por hacer parte del grupo de nodos elegidos, por lo que cualquier accion maliciosa hará que el nodo pierda su reputación y quede excluido(Ocariz Emiliano B., 2019). A diferencia de la minería en *PoW*, estos nodos no ejecutan procesamientos complejos, por lo que hay un mejor uso de los recursos.

Por otro lado, de manera adicional e independiente a los nodos *testigos*, existe otro conjunto de nodos, denominados ***delegados***, cuya principal función es el mantenimiento del rendimiento de la blockchain, proponiendo cambios incluso en algunas variables del mecanismo de validación.

Para finalizar, se pueden destacar las ventajas del mecanismo de consenso *DPoS* en términos de su rápidez comparado con *PoW* y con la mayoría de los mecanismos de consenso; y debido a los bajos requerimientos computacionales, es más eficiente en el consumo de energía.

En la siguiente tabla se pueden observar algunas de las principales cripto-monedas existentes (más importantes o más usadas en el mercado) y el método de consenso usado.

	PoW	PoS	DPoS
Criptomoneda	Bitcoin Ethereum (se encuentra en transición a PoS) LiteCoin Monero Dash Dogecoin	Tezos Dash Cardano Algorand	Cardano Lisk Solana

Tabla 1: Mecanismos de consenso vs criptomonedas

3.4 Permisos en blockchain

Las redes blockchain pueden tener configuraciones de permisos en diversos niveles de acceso, por ejemplo, las configuraciones autorizadas indican acceso solo por invitación, privado o restringido.

Las tres (3) principales configuraciones de permisos en la red blockchain, son las siguientes:

- **Acceso de lectura:** capacidad para ver transacciones e información
- **Acceso de escritura**: capacidad para enviar transacciones e información
- **Acceso a la participación por consenso**: capacidad para servir como un nodo de validación de transacciones.

4. Atributos y beneficios del Blockchain

La tecnología blockchain cuenta con indudables atributos y beneficios para las operaciones y la gestión de procesos en las organizaciones. Algunas de sus características son únicas, y permiten generar confianza digital, así como eficiencia y eficacia en el desarrollo de los procesos organizacionales. A continuación, se presentan algunos de los atributos que caracterizan esta tecnología, no sin antes aclarar que el uso de una blockchain privada o federada puede afectar el alcance de algunos de los atributos:

- ***Inmutabilidad de los registros***: dada la naturaleza de esta tecnología, los datos permanecen almacenados de forma cifrada o encriptada e irreversible (no se puede alterar o cambiar), ya que toda la red tiene una copia de todos los registros y se requiere validar por consenso. Sin embargo, cabe destacar que el grado de inmutabilidad de los registros es directamente proporcional al número de nodos independientes que existan en la red.

- ***Seguridad de la Información***: Se evita la pérdida o alteración de datos, debido a que todos los nodos de la red tienen una copia de los registros, por lo tanto, existe consistencia de la información entre todos los nodos, no hay forma de cambiarlos y no es posible emitir una nueva versión de un registro ya existente que cumpla con los criterios de validación por la red de nodos.

  Además, los datos que se agregan a la cadena son firmados digitalmente, se almacenan de forma cifrada y la interacción con estos se da a partir del consentimiento de su titular

con relación a los datos personales (sin requerir una autoridad central validadora), generando la adecuada trazabilidad de dichas interacciones, quedando registradas en la blockchain.

- ***Trazabilidad:*** una de las características más importantes de la tecnología blockchain es la posibilidad de conocer la traza completa de un elemento de información desde el origen de la primera transacción realizada en la red blockchain, dada la inmutabilidad de los registros y almacenamiento de los eventos que suceden con un elemento almacenado en la blockchain, de manera que se puede conocer con claridad el histórico de la cadena de información contenida en un bloque.

- ***Encadenamiento:*** No se puede editar ni borrar un bloque, únicamente se pueden agregar nuevos bloques, lo que refuerza la seguridad y la trazabilidad de la blockchain.

- ***Base de Datos Descentralizada:*** otra de las características distintivas de blockchain frente a otras tecnologías de gestión de la información, es que las bases de datos están descentralizadas al no estar en un único servidor, lo cual proporciona que los datos no se pierdan (resiliencia a los datos).
  Lo anterior, se logra dado que no dependen de un registro único como fuente de información. Los registros distribuidos permiten que la inmutabilidad de los datos y la confiabilidad del sistema sean únicas, sobre las bases de datos convencionales.

- ***Transparencia:*** Toda la información en blockchain es pública, no puede ser modificada y es fácilmente auditable. Cualquier persona tiene el potencial de controlar el acceso a los registros personales y saber quién los ha accedido. En redes privadas se puede restringir el acceso a cierto tipo de usuarios.

- ***Desintermediación o eliminación de intermediarios:*** Los usuarios de la blockchain pueden interactuar directamente, sin la necesidad de terceros que validen sus transacciones (como bancos y sociedades colectivas de las transacciones), eliminando así la fricción y las demoras asociadas a la intervención de más actores en un proceso, así como los costos de transacción y los riesgos asociados a la presencia de intermediarios.

- ***Certidumbre:*** La blockchain mediante smart contracts, ha definido previamente reglas que son autoejecutables y se activan cuando se cumplen esos criterios predefinidos.

Para finalizar, se destaca la afirmación de Don y Alex Tapscott, quienes indican que el blockchain garantiza confianza al punto de permitir que individuos que no confían entre sí, puedan interactuar de una forma segura sin un intermediario confiable, es este nivel de confianza el valor agregado de esta tecnología, dado que se pueden realizar transacciones rápidamente sin los costos que implica la intervención de un agente externo para realizar la fiscalización y control de la red (Tapscott, Don, Tapscott, Alex, 2019).

5. Tipos de tecnología blockchain de acuerdo a su tipo de acceso.

Según Preukschat (2017) existen tres tipos de blockchain: pública, privada y de consorcio. A continuación, se enuncian aspectos relevantes de cada una de ellas:

5.1 Blockchain Pública

Es una red *descentralizada* de ordenadores que utilizan un protocolo común asumido por todos los usuarios y que les permite registrar transacciones en el libro mayor o ledger de la base de datos, es decier, que les permite generar bloques en la blockchain (Morabito, 2017). Esas transacciones son inalterables, se pueden verificar de forma independiente pero la aceptación de los cambios en los registros, se realiza por consenso entre los participantes de la misma blockchain. En la figura 5 se representa una blockchain pública.

Como sugieren (Bashir, 2017b) y (Preukschat Carlos Kuchkovsky et al., 2017), estas cadenas de bloques no son propiedad de nadie y están abiertas al público para que cualquiera pueda convertirse en usuario, acceder y participar libremente como nodo en el proceso de toma de decisiones (leer, enviar transacciones o participar en el proceso de consenso).

Son *descentralizadas*, en cuanto a que no existe un usuario que tenga mas poder que otro en la red y todos los nodos son iguales entre sí. Son *pseudoanónimas*, debido a que los propietarios de transacciones no son identifcables personalmente, pero sus direcciones sí son rastreables debido a su carácter público (Preukschat Carlos Kuchkovsky et al., 2017) .

Se los considera *"sin permiso"* (en inglés, permisionless), los usuarios mantienen una copia del ledger en sus nodos locales y utilizan un mecanismo de consenso para tomar una decisión sobre el eventual estado del ledger. Bitcoin, Litecoin y Ethereum son ejemplos destacados de blockchain públicos.

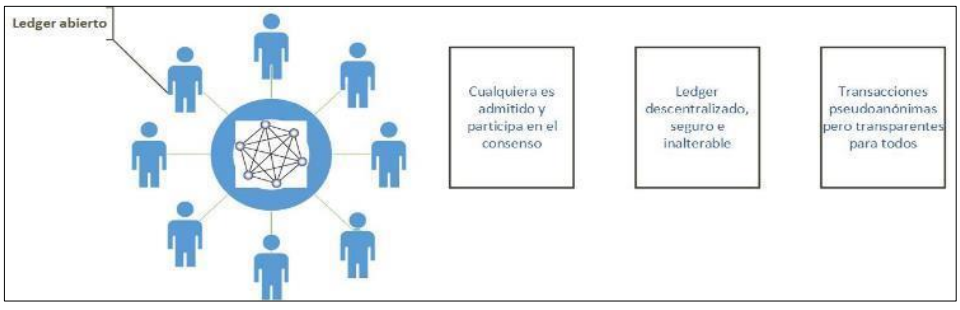


Figura 5. Blockchain pública

5.2 Blockchain Privado

Controlados por una única organización, empresa o entidad que otorga los accesos de edición y el derecho de crear nuevas transacciones en un grupo preseleccionado de usuarios o de nodos (Drescher, 2018), además de controlar los servidores físicos que soportan la red (Ocariz Emiliano B., 2019). También se le considera privada porque no todos los datos

inscritos en la blockchain tienen difusión pública, usualmente por razones regulatorias o de confidencialidad, por lo que sus usuarios pueden acceder y consultar algunas o todas las transacciones realizadas.

La blockchain privada es *cerrada* porque solo las personas o entidades invitadas a participar pueden registrar las transacciones. Para lo anterior, el protocolo predeterminado podrá incluir distintos niveles de acceso a los usuarios según sean los fines pretendidos, sin embargo Ocariz afirma que "su seguridad ya no está garantizada tanto por el mecanismo sino por la protección que desarrolle el gestor de la red" (Ocariz Emiliano B., 2019), por lo anterior, la blockchain privada puede establecer el nivel de *anonimato* que quiera para realizar o proteger transacciones.

Mientras la blockchain pública es descentralizada porque no se controla quien participa en la misma, la blockchain privada es *distribuida*, en el sentido de que es una base de datos repartida en varios nodos, es decir que el número de nodos de lo que se componga puede estar limitado al número de participantes que han sido invitados a participar. Por lo anterior, también reciben el nombre de "blockchain de acceso basado en permiso" (en inglés, permissioned).

Hyperledger, Ripple y R3 son ejemplos representativos de blockchain privado y son habituales en el mundo empresarial.

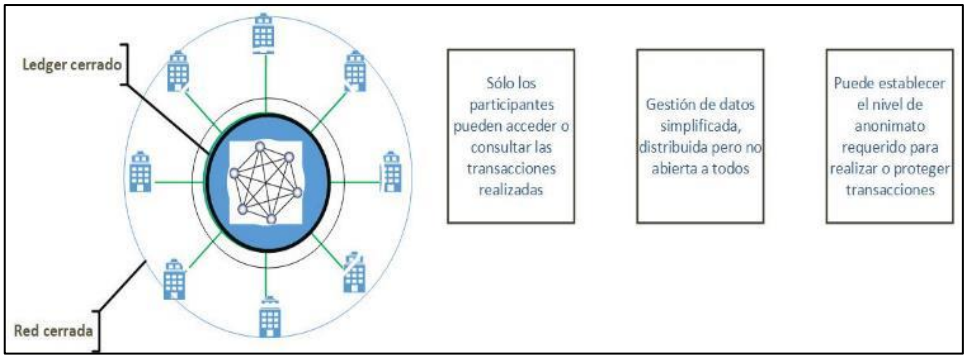


Figura 6. Blockchain privada

5.3 Blockchain Federada o de Consorcio

Es una blockchain intermedia que trata de aprovechar las ventajas de la blockchain pública y la privada, por lo que también se le conoce como blockchain híbrida o blockchain semiprivada.

Este tipo de blockchain no es propiedad de una sola empresa, sino de un grupo de ellas, por lo que en el proceso de consenso deben acordar reglas y definir representantes dentro de la red, por lo que sus integrantes pueden decidir quiénes participan (operaciones en modo privado) y qué transacciones se hacen públicas (registro en la red pública) (Preukschat Carlos Kuchkovsky et al., 2017).

El software que respalda la blockchain federada es abierto, por lo que la comunidad desarrolladora puede reutilizar el código, programando la cadena de manera personalizada decidiendo quiénes participan y bajó qué reglas se regulan las transacciones (Bambara, 2018)(Morabito, 2017).

Los blockchain de consorcio se consideran “blockchain autorizados”, no tienen un modelo de participación abierto al público, sino que requiere de la aprobación del consorcio y el proceso de consenso está en muchos casos ligado a la votación de cada nodo representante de cada entidad que conforma el consorcio.

Aunque no esté verdaderamente descentralizado, este tipo de blockchain autorizado sigue ofreciendo integridad, transparencia y seguridad, siendo atractivo para los casos de uso business-to-business (negocios dirigidos a empresas). Evernym, BigchainDB y Diaspora, son ejemplos de blockchain de consorcio.

En el cuadro que sigue se presenta brevemente la diferencia entre los tipos de blockchain descritos en este numeral:

Arquitectura/ Característica	**B. pública**	**B. privada**	**B. de consorcio**
Acceso	• Cualquiera • permisionless	• Una única organización • Permissioned	Varias organizaciones (seleccionadas)
Participantes	• Sin permisos • Anónima • No hay administradores	• Con permisos • Identidades conocidas • Único administrador	• Con permisos • Identidades conocidas • Hay más de un administrador
Seguridad	Mecanismos de consenso descentralizados (PoW, PoS)	Algoritmos de consenso propios (Transacciones verificadas dentro de la organización, es decir, centralizadas)	Algoritmos de consenso propios (varios nodos preseleccionados dentro de las organizaciones, es decir, parcialmente descentralizadas)
Velocidad transaccional	Lenta	• Ágil • Rápida	• Ágil • Rápida
Implementación de smart contracts	Se puede	Se puede	Se puede
Ejemplos	• Bitcoin • Ethereum	• Hyperledger • R3	• Evernym • BigchainDB

Tabla 2: Diferencias entre los tipos de blockchain

6. Arquitecturas de Blockchain

Blockchain cuenta una arquitectura que se debe considerar en la adopción de una solución, debido a sus posibilidades de implantar redes masivas, distribuidas, descentralizadas y seguras, debido a que el propósito es prescindir de la entidad central que administre la red, como se menciona en apartados anteriores, por ello se tiene que los registros de la blockchain se ejecutan mediante bases de datos descentralizadas y distribuidas en las que la información se almacena en varios servidores. Lo anterior, en contraposición con las bases de datos o registros de información centralizados que son las redes en las que los datos se almacenan en un único servidor, aunque sea accesible desde otros lugares y por diferentes entidades, de este modo, blockchain en no será en absoluto la mejor opción.

A continuación, se desarrolla una breve exposición sobre los tipos de arquitectura de

redes:

6.1 Descentralizada

Una arquitectura *descentralizada* consiste en tener una estructura de nodos donde la información funciona tipo árbol. Desde el centro se emiten informaciones y esas informaciones son recibidas por unos nodos intermedios, de tal forma que esos nodos intermedios pueden o no emitir esa información hacia los receptores finales (De Ugarte, 2018)

En una red descentralizada no existe un único nodo central, hay un centro colectivo de diversos puertos de conexión, es decir, todos los nodos se conectan entre sí, sin que tengan que pasar obligatoriamente por uno a varios centros por lo que no hay un punto único de decisión, a su vez cada nodo toma la decisión que más le conviene en función de las reglas de consenso que el operador del nodo ha elegido. Adicionalmente, como no se depende de un solo servidor, no adolecen de un punto de fallo, por lo que son más estables y más rápidas que las bases de datos centralizadas.

Los nodos en un sistema descentralizado no tienen conocimiento del estado de la totalidad del sistema, pero toman las decisiones que más le convienen con la información que tienen. Este tipo de red se rige por el principio de la adhesión o la participación.

6.2 Distribuida

Una red de datos ***distribuida*** funciona como una única red de datos lógica, instalada en una serie de ordenadores (nodos) ubicadas en diferentes lugares geográficos y que no están conectadas a una única unidad de procesamiento, pero sí están totalmente conectadas entre sí para proporcionar la integridad y accesibilidad a la información desde cualquier punto. Esta arquitectura ha sido diseñada para eliminar la centralización quitando la dependencia de un servidor.

En este tipo de arquitectura, todos los nodos contienen información y todos los clientes del sistema están en condición de igualdad (*Redes Centralizadas VS Distribuidas. | by ICommunity.Io | Medium*, n.d.). Asimismo, la escalabilidad es mayor que en las redes centralizadas dado que los nodos distribuidos pueden soportar la carga de tráfico ya que esta se comparte entre varios ordenadores, no obstante, el reto en cada uno de éstos es mantener el almacenamiento requerido para soportar el tráfico progresivo y en aumento. A su vez, este tipo de redes permite suministrar servicios a una mayor velocidad, dado que es menos probable que sucedan “cuellos de botella” debido que es posible soportar el tráfico desde cualquiera los nodos.

En un sistema distribuido, el procesamiento se comparte entre múltiples nodos, pero las decisiones son centralizadas y tienen conocimiento del estado total del sistema. Por lo tanto, la conexión entre los diferentes nodos es P2P, en lugar de usar un nodo central.

En este tipo de arquitectura, todos los datos se distribuyen entre los nodos de la red. Si se agrega, edita o elimina un dato en cualquier servidor de la red, se reflejará en los demás servidores de la red. Si se aceptan algunas enmiendas legales, se difundirá nueva información entre otros usuarios de toda la red; de lo contrario, los datos se copiarán para coincidir con

los otros nodos. Por lo tanto, el sistema es autosuficiente y autorregulador.

Dado que la cantidad de ordenadores en la red distribuida es grande, ataques de denegación de servicios (DDoS por sus siglas en inglés) son posibles solo en caso de que su capacidad sea mucho mayor que la de la red.

Comprender los sistemas distribuidos es esencial para comprender blockchain porque básicamente blockchain en su núcleo, es un sistema distribuido descentralizado.

6.3 Centralizada

La arquitectura ***centralizada*** está gestionada por un solo nodo y sus usuarios pertenecen a la misma comunidad, esto es, donde todos sus nodos son periféricos, y se conectan a uno central y para acceder a la información se debe acceder al nodo principal del sistema, conocido como *servidor* (*Redes Centralizadas VS Distribuidas. | by ICommunity.Io | Medium*, n.d.). En este tipo de redes, la caída del nodo central corta el flujo de datos a todos los demás nodos.

Se utiliza principalmente en servicios web, alojadas en un servidor centralizado por el que tienen que pasar todas las personas que quieran acceder a ella, por ejemplo: Github y Airbnb.

En una arquitectura centralizada, todos los clientes están conectados al servidor. Por lo tanto, todas las solicitudes para recibir, cambiar, agregar o eliminar datos pasan por un ordenador principal.

En consecuencia, es capaz de llevar a cabo su trabajo de manera efectiva solo para el número específico de participantes. Si la cantidad de clientes es mayor, la carga del servidor puede exceder el límite de disponibilidad durante el tiempo en que se presente dicho incremento.

Las redes centralizadas son difíciles de escalar ya que la capacidad del servidor puede ser limitada y el tráfico no puede ser infinito.

La figura 7, representa gráficamente tres formas distintas de organizar una red: centralizada, descentralizada y distribuida.

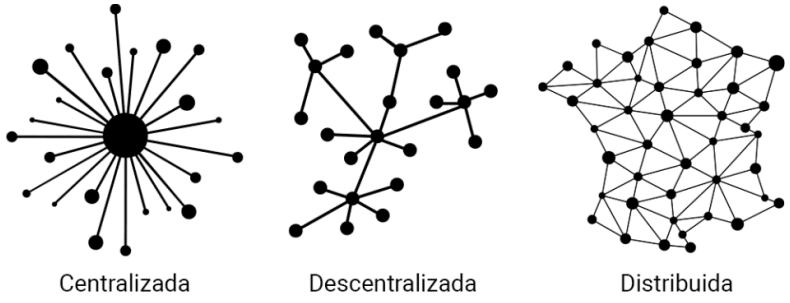


Figura: 7. Topología de redes. Fuente: (Baran Paul, 1964)

7. Campos de aplicación de Blockchain

Como se mencionó al inicio de este documento, blockchain se popularizó como la tecnología detrás de la criptomoneda Bitcoin, no obstante, su potencial supera ampliamente

la del intercambio de valor de criptoactivos y se ha usado en aplicaciones para conservar la trazabilidad de registros o de activos, por ejemplo, en temas inmobiliarios, financieros, registros que requieran la identidad de la persona u organización, aplicaciones en logística, cadena de suministro, o cualquier tipo de cosa o título valor y se acerca de una manera accesible a soluciones para diferentes industrias y sectores, incluyendo el modelo de internet del valor, como una herramienta para gestionar y compartir el valor de activos o bienes digitales sin la necesidad de depender de una entidad central de confianza (Triana Casallas et al., 2020), en escenarios como internet de las cosas-IOT y big data (Dolader et al., 2017).

A continuación, se presentan algunos casos en los que se han encontrado perpectivas de aplicación de blockchain o en los que ya se han abordado soluciones a partir de esta tecnología:

Temas	Soluciones
Manejo, validación y registro de la información	• Mantenimiento de registros • Registro y protección de la propiedad Industrial e Intelectual • BlockChain como mecanismo para proveer Seguridad, Transparencia y Confianza
Red de pago BlockChain	• Automóviles con conectividad para realizar micro pagos • Drones de entrega aérea que permiten realizar micro pagos
Desarrollos y herramientas en el BlockChain	• Plataformas para el desarrollo de Aplicaciones BlockChain • Plataforma y blockchain Ethereum • Contratos inteligentes • Identidad Digital
Aplicaciones con características descentralizadas	• DAO (Organizaciones Autónomas Descentralizadas), DAC (Corporaciones Autónomas Descentralizadas) y DAS (Sociedades Autónomas Descentralizadas) • Dapps (Aplicaciones Distribuidas/Descentralizadas)
Trazabilidad	• Ciclo de vida del producto • Cadena de Suministro -Sistemas de trazabilidad y Logística • Trazabilidad en el sector de la automoción • Transparencia y Trazabilidad en el Sector de Alimentos
Aplicaciones en el área de la tecnología y la computación	• RFID (Identificación por radiofrecuencia) • Inteligencia Artificial soportada en BlockChain • Aplicaciones que incorporan IoT (Internet de las Cosas) y BlockChain • Comercio Electrónico • M2M (Interacciones de Máquina a Máquina) • CMfg (Fabricación de la nube)
Aplicaciones ambientales	• Blockchain y el Cambio Climático
BlockChain para sectores empresariales y negocios	• BlockChain y Empresas Aseguradoras • Intercambio de electricidad por medio acuerdos soportados en BlockChain • Proyectos de construcción • Intercambio y Transacción de conocimiento entre equipos y organizaciones • Economía colaborativa y Compartida • Blockchain y Procesos de negocio – BPM (Business Process Management)
Sector público/Gobierno	• Voto electrónico • Registro de tierras y transferencia de la propiedad • Identidad digital

Tabla 3: Temáticas abordadas con blockchain para soluciones en servicios, industria y logística. Elaboración propia a partir de: (Rodríguez Molano et al., 2020)

Ahora bien, en seguida se presentarán con algo más de detalle algunas de las proyecciones

de blockchain para categorías o casos de uso específicos:

- **Seguridad en Big Data:** Se puede usar la tecnología blockchain para dar seguridad y verificabilidad a entornos de big data, de esta manera la blockchain aporta transparencia, trazabilidad, portabilidad y escalabilidad.

- **Blockchain e IoT:** Con la evolución de la IoT, el número de dispositivos conectados a la red de redes crecerá vertiginosamente, asimismo deberá abordarse el ámbito de la seguridad, especialmente en el nivel de supervisión de los dispositivos, ya que muchos de ellos (los de bajo coste) no se encuentran supervisados con los niveles usados en el mundo de la computación, debido principalmente a que para los fabricantes resulta muy costoso tener que enviar actualizaciones frecuentes a millones de dispositivos. Adicionalmente, para el consumidor, existe gran desconfianza en dispositivos que se comunican con su fabricante sin la supervisión del usuario final.

Ahora bien, este escenario carente de confianza y de baja percepción de seguridad podría solucionarse con una blockchain, de la que se puede aprovechar su transparencia, robustez, fiabilidad y su arquitectura distribuida, de forma que los dispositivos consultarían la blockchain para averiguar si su firmware está actualizado. En caso que no lo esté, pedirían a otros nodos el envio de la nueva versión. Una vez recibida, podrían usar el código de la blockchain para comprobar que el firmware no ha sido alterado en modo alguno, evitando así las intrusiones(Dolader et al., 2017).

- **Causas sociales:**
  Donaciones: Se han desarrollado mecanismos a partir de blockchain para realizar donaciones para niños con carencias, de manera que se pueda establecer y mantener la trazabilidad de las donaciones(Tapscott, Don, Tapscott, Alex, 2019), (Ocariz Emiliano B., 2019)
  Atención a refugiados: Con la información de la persona y su correspondiente registro en la blockchain, esta tendrá acceso a beneficios para aliviar parte de sus carencias, haciendo llegar la ayuda, de forma oportuna y sin corrupción o exceso de trámites(Ocariz Emiliano B., 2019).

- **Educación:** El manejo seguro y abierto de la información académica se podría realizar a través de un sistema basado en blockchain, con la historia académica de los individuos e inclusive puede extenderse al diseño, aplicación y evaluación de pruebas(Ocariz Emiliano B., 2019).

- **Sanidad:** Trazabilidad de la información y obtención de un historial natural de la información de los pacientes de manera descentralizada(*What Is Blockchain for Business? – IBM Blockchain | IBM*, n.d.), que más adelante pueda convertirse en un registro transversal e interoperable de los pacientes, conectando los proveedores de salud(Ocariz Emiliano B., 2019). En este sentido, la pandemia ocasionada por la propagación del coronavirus SARS-CoV-2, ha evidenciado de forma especial las posibilidades que ofrece la blockchain en la gestión de la cadena de suministro de productos sanitarios (aunque esto puede hacerse extensivo a cualquier sector), con el propósito que la blockchain hace haga posible las mejoras en la gestión de inventarios, la automatización de procesos y reducir los gastos generales.

- **Alimentación:** Aplicaciones en la seguridad alimentaria para tener constancia de todo el proceso por el que ha pasado un alimento, al que luego se podría tener acceso mediante un código QR en el producto(*What Is Blockchain for Business? – IBM Blockchain | IBM*, n.d.).

- **Logística y transporte:** Aplicaciones blockchain para la automatización de procesos, para el seguimiento de los bienes transportados con gran precisión, así como para la simplificación de trámites administrativos y controles aduaneros, evitando la falsificación de documentos de importación/exportación(*What Is Blockchain for Business? – IBM Blockchain | IBM*, n.d.).

  Sistema de seguimiento de transportes. Blockchain tiene aplicabilidad en este caso, que permitiría hacer el sistema más simple, más transparente y menos costoso. En primer lugar se unificaría en la blockchain, la información de bases de datos donde las empresas intervinientes en todo el proceso, van actualizando el estado del envío en función de la información proporcionada por las otras empresas o por sus agentes. Con lo anterior, al llegar el envio a un destino parcial o final, se añadiría una actualización a la blockchain. Al estar todas las actualizaciones firmadas con las claves privadas del que entrega y el que recoge y con la clave del envio, esta actualización actuaría como prueba criptográfica de la recepción del envio por parte del destinatario. La transparencia y fiabilidad del concepto ayudaría con la resolución de eventuales disputas entre los participantes(Dolader et al., 2017).

- **Procesos gubernamentales y democráticos:** Se destacan avances en la seguridad e integridad de los datos, elecciones políticas transparentes y públicas, registro de propiedad, etc.

  Registro de propiedades: Se han creado proyectos que intentan resolver mediante blockchain, el problema de titulación de tierras que requiere que de manera confiable, segura y coordinada, se administren los títulos de propiedad y las transacciones derivadas.

  Contratación pública: uno de los ámbitos más sensibles porque incorpora la asignación de gran proporción de los presupuestos y que, de manera acertada, podría beneficiarse con la implementación de la tecnología blockchain en las diferentes etapas de la contratación, dentro de las cuales se encuentra la información de contratistas, para crear un registro descentralizado de ofertas de contratos públicos que permita una valoración automatizada de las ofertas a través de Smart contracts, así como el control a la ejecución contractual. Lo anterior, podría incidir en la correcta aplicación de la legislación, de los compromisos adquiridos por las partes, así como en una mayor transparencia.

  Estado integrado: Uso de estructuras una blockchain única para incluir documentos oficiales (pasaporte, documento de identificación, licencia de conducción, títulos de propiedad, expedientes académicos, situación fiscal y laboral, entre otros) que existen en diferentes bases de datos. Con lo anterior, la blockchain podría ofrecer servicios integrados sin pasar por un procesamiento central (Tapscott, Don, Tapscott, Alex, 2019)

  Votaciones: La votación ha evolucionado y hoy por hoy es un proceso más seguro, sin embargo, aún tiene limitaciones que giran principalmente en torno al fraude y en la falta

de transparencia, lo que sin duda genera desconfianza en el gobierno. Los sistemas de votación basados en blockchain pueden resolver estos problemas porque suministran seguridad de extremo a extremo y transparencia en el proceso. Por un lado, la seguridad se proporciona con la integridad y autenticidad de los votos por cuanto se usa criptografía de clave pública y por otro lado, la inmutabilidad, que garantiza que los votos emitidos no se vuelvan a emitir, sin contar con que se protege la privacidad de los votantes en la blockchain(Bashir, 2017a).

Identidad digital: Autenticación para acceder a servicios seguros, así como encriptar, verificar y firmar documentos digitalmente(Triana Casallas et al., 2020).

CAPÍTULO 4. SMART CONTRACTS

Un uso muy práctico del *blockchain* son precisamente los *smart contracts* (contratos inteligentes. A continuación, se presentará información relevante de estos y su relación con blockchain.

1. Definición y antecedentes

El término de ***Smart contract*** *(contrato inteligente)* es anterior a la blockchain con Nick Szabo, quién en 1994 fue el primero en utilizar el término Smart contract, como un protocolo de transacciones sistematizadas que ejecutan los términos de un contrato (programables con relación a condiciones) (Nakamoto, 2008).

El concepto nace en el marco de la definición del objetivo de lo que se denominaba "prácticas evolucionadas" de la ley de contratos y prácticas comerciales en el ámbito del diseño de protocolos de comercio electrónico, justificaba que la especificación una verificación o ejecución a través de protocolos criptográficos y otros mecanismos de seguridad digital, podrían construir una mejora importante sobre los contratos habituales. Por otro lado, fue hasta 2009 que esta idea de Smart contracts fue implementada, lo hizo BitCoin, donde las transacciones de bitcoin se podían usar para transferir el valor entre usuarios (Bashir, 2017a), a través de una blockchain.

Szaboo describió los Smart contracts de la siguiente manera: *"Un contrato inteligente es un protocolo de transacción computarizado que ejecuta los términos de un contrato. Los objetivos generales son satisfacer condiciones contractuales comunes (tales como condiciones de pago, gravámenes, confidencialidad e incluso cumplimiento), minimizar las excepciones tanto maliciosos y accidentales, y minimizar la necesidad de intermediarios de confianza. Los objetivos económicos relacionados incluyen reducir las pérdidas por fraude, los arbitrajes y los costos de ejecución, y otros costos de transacción"* (Bashir, 2017a), (Ocariz Emiliano B., 2019).

Otra definición de *Smart contract* que puede ser de interés consiste en entenderlo como cualquier contrato que se ejecuta por sí mismo automáticamente, de forma independiente sin la medición de terceros, se escriben como programas o códigos informáticos en los que se definen y describen reglas y consecuencias, programables con relación a sus funciones mediante lógica matemática (IF+Then) (Triana Casallas et al., 2020), es decir, *si* pasa esto, entonces *ocurrirá* eso. Lo anterior, en lugar de ser escritos en lenguaje legal sobre documentos impresos como tradicionalmente se conocen (Ocariz Emiliano B., 2019), toda vez que actúan como acuerdos vinculantes entre dos o más partes, son imparables y automáticos, pueden ser ejecutadas por condiciones externas (Triana Casallas et al., 2020) y residen en la blockchain.

Es importante aclarar que un Smart contract es un concepto asociado al diseño del código de programación en una blockchain, totalmente diferente a los contratos tradicionales, por lo que no se constituyen como contratos desde el punto de vista legal y jurídico, toda vez que responden a una lógica de instrucciones que rigen transacciones comerciales o acuerdos entre partes que se ejecutan automáticamente y se almacenan en la blockchain (Gupta, 2017), extendiendo así la utilidad de está última, de mantener un registro de entradas y

trazabilidad de transacciones, para pasar a aprovechar la capacidad de la blockchain a la realización de otros cálculos adicionales que se ejecutan de manera autónoma distintos de la transferencia de monedas (Makhdoom et al., 2019).

El ejemplo clásico utilizado para demostrar contratos inteligentes en forma de código que se ejecuta automáticamente es una máquina expendedora ya que toma el dinero y dispensa un producto y el cambio correcto en función del precio de compra, a diferencia de una persona, la máquina se comporta algorítmicamente; se seguirá el mismo conjunto de instrucciones cada vez en todos los casos(Ocariz Emiliano B., 2019),(Swan, 2015).

A continuación, se presentan las características de los Smart contracts:

- *Autónomos:* Una vez se da la ejecución del Smart contract, no se necesita de un intermediario para firmarlo o para hacer cumplir las reglas allí establecidas debido a que cuentan con cláusulas generales pre validadas.(De Filippi, 2014), (Vass Company, 2017).
- *Confiables:* Los documentos son encriptados y validados por varios agentes, lo que hace casi imposible que se puedan perder o cambiar, por ello las partes del contrato no tienen que confiar una en la otra, para que se dé cumplimiento a las claúsulas establecidas en el contrato (Vass Company, 2017).
- *Autosuficiencia:* en su capacidad para reunir recursos, es decir, recaudar fondos mediante la prestación de servicios o la emisión de acciones y gastarlos en los recursos necesarios, como en la potencia de procesamiento o en el almacenamiento (De Filippi, 2014)
- *Seguros:* Garantizan el encriptado y la misma información se valida en varios bloques a la vez.
- *Rápidos*: Son autoejecutables, por ello hay inmediatez en las transacciones.
- *Exactos:* Lo que se ejecuta es exactamente lo que dice el contrato, no hay errores al evitar que intervengan personas para llevar a cabo las acciones que conllevan (Vass Company, 2017).
- *Descentralizados:* Los Smart contracts no permanecen en un único servidor, sino que se distribuyen y se ejecutan automáticamente en los nodos de la red (Swan, 2015), (De Filippi, 2014)
- *Deterministas:* Los Smart contracts son deterministas dado que la misma entrada genera la misma salida(Christidis & Devetsikiotis, 2016), es decir, que cuando se ejecuta no requiere información desde fuera de la blockchain, de lo contrario ocasionaría problemas para la obtención de consensos debido a que depende de información desde una fuente externa a la blockchain, sin embargo, como solución, el Smart contract tendría que aplicar oráculos(Morabito, 2017).

Para finalizar, los Smart contracts que utilizan atributos de blockchain, pueden desarrollarse para garantizar que:

- Las transacciones sean auditables
- Los activos se puedan verificar a través de la cadena de custodia
- Los registros de transacciones no sean alterados

2. Estructura y funcionamiento de los Smart contracts

2.1 Estructura de los smart contracts

Un Smart contract está definido por el código (con las instrucciones if+then) y lo ejecutado

(o aplicado) por el código, automáticamente sin discreción una vez se cumplan las condiciones especificadas. Básicamente, un Smart contract está conformado por un saldo de cuenta, un almacenamiento privado y un código ejecutable (Alharby & van Moorsel, 2017).

El estado del Smart contract se almacena en la blockchain y se actualiza cada vez que se invoca el contrato y está comprendido por el saldo de cuenta y el almacenamiento (Alharby & Moorsel, 2017). Una vez que el Smart contract se crea en blockchain, su código no se puede cambiar y es identificable con una dirección única asignada de 20 bytes(Luu et al., 2016).

Para ejecutar un contrato, los usuarios pueden simplemente enviar una transacción a la dirección del contrato. Esta transacción luego será ejecutada por los nodos mineros de la red para llegar a un consenso sobre su salida (Alharby & van Moorsel, 2017).

En la figura 8 se puede evidenciar la estructura del sistema de smartcontracts:

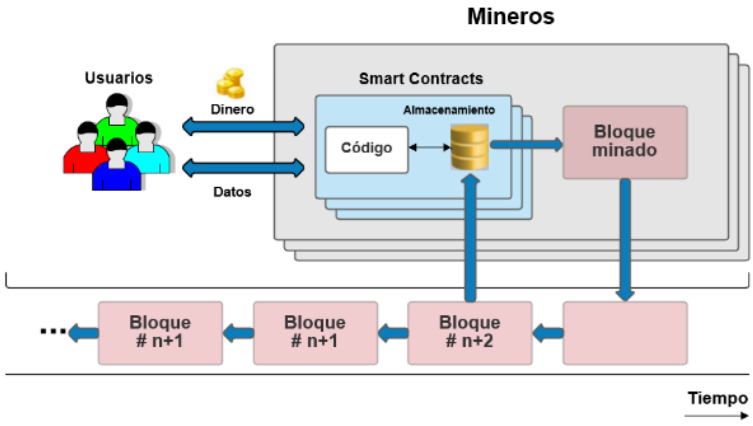


Figura: 8. Estructura del sistema de smart contracts Fuente: Adaptada de (Alharby & van Moorsel, 2017)

2.2 Funcionamiento de los smart contracts

Si bien, en los Smart contracts se definen las reglas en torno a un acuerdo, en estos también radica la capacidad para hacer cumplir esas obligaciones de forma automática, por ello, cuando un evento ocurre, como puede ser, el vencimiento de una fecha, un resultado esperado o precio de un bien alcance cierto límite, el contrato es ejecutado de acuerdo a los términos específicados en el código de software de una manera transparente y sin conflictos, a la vez que evita los servicios de intermediarios (Rosic, 2017)

Los Smart contracts pueden ser activados a través del envío de transacciones que cumplen las reglas que rigen el contrato(Makhdoom et al., 2019). Esas transacciones pasan por tres fases: la de las entradas, del intérprete del contrato y las salidas, como se muestra en la figura 9.

- ***Entradas***: en esta fase se especifican:
  - Un ID (hash), que al igual que en el caso de las transacciones, se calcula a partir de los demás datos. Es útil para evitar que se manipule algún dato del contrato antes de ser ejecutado. (Stefanescu, 2019)
  - Condiciones del contrato (puede ser fecha en la que se debe ejecutar, cantidades a enviarse, etc.) (Stefanescu, 2019)
  - Solicitud de transacción, las dependencias que puedan existir y el estado actual del libro mayor (Hyperledger, 2018b).

• ***Interpretación del contrato***: en esta fase se inicia con el estado actual del libro mayor y el código de Smart contract, recibe la solicitud de transacción y esta se comprueba, en seguida se procede con su aprobación o rechazo, según sea el cumplimiento de las condiciones del contrato.

Según la solicitud de transacción, el contrato puede: leer o escribir en su almacenamiento privado, almacenar dinero en el saldo de su cuenta, enviar o recibir mensajes o dinero de usuarios o de otros contratos e inclusive puede crear nuevos contratos(Alharby & van Moorsel, 2017).

• ***Salidas:*** en esta fase, las salidas son diferentes según la situación que ocurra con la validación de la transacción en la fase anterior:

- Solicitud válida: Las salidas incluyen un nuevo estado, una declaración de corrección y cualquier sugerencia de pedido requerida para los servicios de consenso para el compromiso final con la blockchain.
- Solicitud no válida: Las solicitudes no válidas se rechazan del sistema.

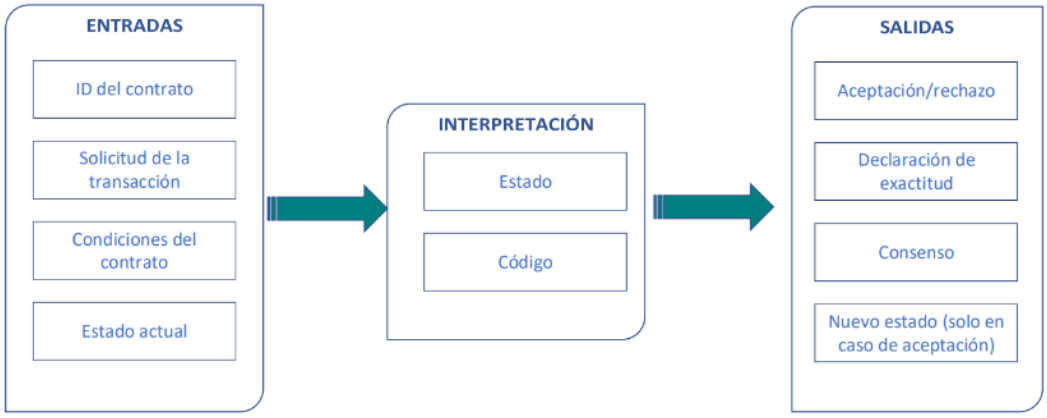


Figura: 9. Fases para la ejecución de un smart contract.

Ahora bien, llevando la explicación anterior acerca de la estructura y funcionamiento de un smart contract a la lógica empresarial, se obtendría un acercamiento con el diagrama de flujo que se presenta enseguida:

Figura: 10. Estructura de aplicación de la lógica empresarial con los smart contracts. Fuente: (BBVA Research, 2015)

Los Smart contracts pueden estar codificados de modo que reflejen cualquier tipo de lógica empresarial basada en los datos: desde acciones tan sencillas como votar por una publicación en un foro hasta acciones con un mayor nivel de complejidad, como garantías de préstamos y contratos de futuros, así como acciones sumamente complejas como la fijación de prioridades de pago en una nota estructurada.

3. Plataformas de Smart contracts

Dadas las características de inmutabilidad, confidencialidad, trazabilidad y transparencia de Blockchain junto con la automatización que implican los Smart Contracts, se requiere identificar las plataformas de desarrollo que permitan la incorporación de estos últimos a la blockchain.

A continuación, se presentan algunas de las plataformas más importantes que tienen gran influencia en la capitalización del mercado mundial y que además cuenta con solución en producción para realizar Smart contracts:

3.1 Ethereum

La plataforma Ethereum fue propuesta por Vitalik Buterin en 2013, se desarrolló a través de financiamiento colectivo ICO (initial coin offering) e inició su funcionamiento a mediados del 2015. Posiblemente es la más conocida de todas las plataformas; después de bitcoin, es el segundo proyecto de blockchain más valioso por capitalización de mercado y es el primer blockchain que desde el inicio fue diseñado para usar smart contracts.

Ethereum es una plataforma descentralizada y basándose en los smart contracts desplegados en la red se pueden crear aplicaciones conocidas como Distributed Application (DApp). Ethereum se basa en el modelo blockchain con tecnología de contabilidad distribuida (DTL) y una de sus funciones de mayor relevancia consiste en la ejecución de los smart contracts.

Una de las características de Ethereum es que es una red de carácter público. Cualquiera puede unirse a la red manteniendo un nodo, y la información es pública para todos, por lo que sería una elección adecuada para aquellos contratos de dominio público, y en particular aquellos smart contracts para regular relaciones *Business-to-Consumer.*

Su algoritmo de consenso trabaja mediante el mecanismo de prueba de trabajo (PoW) que mediante una red basta de minero las transacciones se validan y verifican para ser agregadas a un bloque. Si bien, PoW es uno de los mecanismos más populares y seguros dentro de esta tecnología, sus propias características impactan negativamente en la performance y en los recursos que necesita debido a que al tener que realizar pruebas criptográficas complejas, los tiempos de creación de nuevos bloques son altos, así como el consumo energético.

La máquina virtual de Ethereum (EVM) es quien se encarga de ejecutar los smart contracts sobre la Blockchain cuya característica principal es que es “turing-complete”, es decir, puede codificar cualquier cálculo que pueda llevarse a cabo. A su vez, integra un lenguaje de programación de alto nivel denominado, Solidity. Este lenguaje es parecido a Javascript en la sintaxis, lo que facilita su adopción, pero se encuentra enriquecido con conceptos de

programación orientada a objetos como C++ y Python; de hecho, con Solidity se pueden usar un conjunto de operadores e instrucciones que permiten ejecutar programas similares a cualquier lenguaje de programación moderno, por ejemplo LLL y Serpent.

Ethereum cuenta con su propio token, una criptomoneda llamada Ether, la cual permite impulsar la plataforma en la creación de smart contracts, aplicaciones distribuidas o invocar sus métodos, en últimas, es utilizada para pagar los costes asociados a las transacciones en la plataforma.

Cada transacción que es enviada a la red requiere cierta cantidad de Gas (coste asociado a realizar operaciones en la red, cuyo valor se define dependiendo de la potencia y complejidad computacional que es necesaria para procesar dicha transacción).

No obstante, el Gas no debe ser visto como un llanamente como un coste, debido a que también aporta una serie de ventajas a la red de Ethereum(Prusty, 2017):

- *Optimiza los contratos*: al suponer un gasto, evita que la red se llene de cálculos innecesarios que la harían menos eficiente. También ayuda a que la cadena sea más ligera, por cuanto no se llena de información no relevante.

- *Impide el mal uso de la red:* el uso de Gas hace inviable la utilización de la red para realizar Spam.

- *Asegura la red ante programas maliciosos:* el uso de Gas finito protege la red ante el uso de programas maliciosos que pudiesen ejecutar ataques a la red mediante el uso de bucles infinitos y exceso de computación.

- *Garantiza el mantenimiento de la red*: el Gas se usa para recompensar a los mineros por mantener el nodo y realizar los cálculos necesarios.

La principal ventaja de usar Ethereum para ejecutar smart contracts es que facilita la interacción entre ellos. Además, no tiene que preocuparse por integrar protocolo de consenso y otras cosas; en su lugar, solo necesita escribir la lógica de la aplicación.

- *Los programas son inalterables.* Una vez desplegados en la red, no se pueden eliminar. En caso de necesitar la modificación de un smart contract, se debe desplegar una nueva versión y la aplicación deberá invocar este nuevo contrato.

  La modificación de un Smart contract podría mejorar la calidad general al permitir la corrección de errores o incorporar mejoras en Smart contracts desplegados, sin embargo, al brindar la posibilidad de modificar el smart contract ya desplegado, se agrega vulnerabilidad al sistema porque este recurso puede ser usado de forma maliciosa. Por este motivo, en caso de permitirse, deberán ser evaluados los mecanismos utilizados para ejecutar la actualización.

3.2 TRON

TRON es una plataforma descentralizada de contratos inteligentes encargada de otorgar un desarrollo más óptimo respecto de Bitcoin y Ethereum, para la ejecución y escalabilidad de aplicaciones descentralizadas (Dapps).

TRON fue fundada por Justin Sun a mediados de 2017 y de acuerdo con su documento

técnico(Tron Fundation, 2018), la base del código TRON fue originalmente extraído de una bifurcación (fork) de Ethereum y utiliza una copia del lenguaje de smart contract de Solidity para la configuración de sus contratos. Por lo anterior, los smart contracts y los estándares de token de Ethereum son compatibles con TRON.

Esta plataforma adopta una arquitectura de tres (3) capas: almacenamiento, núcleo y aplicación como lo muestra la figura 11, donde además se puede ver que los Smart contracts hacen parte de uno de los módulos del núcleo, además del mecanismo de consenso, que es la prueba de participación delegada (DPoS), dónde veintisiete (27) Súper Representantes electos producen bloques para la red, lo que le ha permitido a TRON, lograr tasas de transacciones mucho más rápidas que otras blockchain. Es importante aclarar que, para comunicarse entre las distintas capas, TRON utiliza el protocolo Google Protocol Buffers (lo que da soporte multi lenguaje).

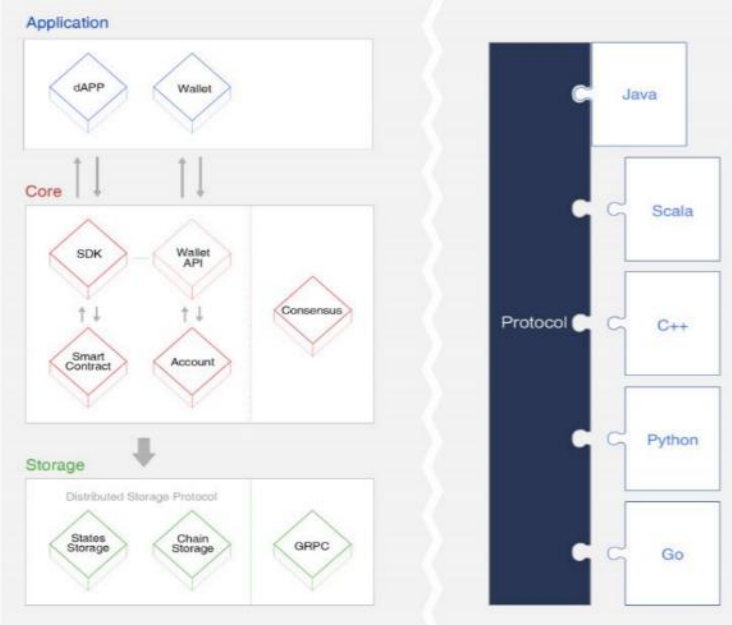


Figura: 11. Arquitectura de Tron. Fuente: (Tron Fundation, 2018)

Desde el punto de vista del almacenamiento, en esta capa, TRON implementa un protocolo único distribuido, que consiste en dos componentes diferentes: por un lado, el almacenamiento de los bloques de la cadena y por otro, el almacenamiento del estado.

Para poder realizar transacciones o crear Smart contracts en TRON, es necesario contar con dos tipos de recursos: ancho de banda (bandwidth) y energía.

Por un lado, el ancho de banda es el recurso que permite a los usuarios participar en la elección de los nodos super representativos y poder enviar transacciones a la red sin necesidad de pagar por ello; por el otro, la energía es el recurso necesario para poder ejecutar un smart contract (similar al Gas de Ethereum). Para generar alguno de los recursos (ancho de banda o energía) el usuario debe, a través de su billetera electrónica, congelar una cierta cantidad de TRX (criptomoneda de TRON) por un periodo de tres (3) días. Cuantos más TRX se congelen, mayor será la cantidad de ancho de banda o energía que tenga el usuario.

En Tron, generalmente no se cobra para la mayoría de las transacciones y cualquier transacción de consulta es gratis, es decir, no cuesta energía ni ancho de banda.

3.3 Hyperledger fabric

Hyperledger fabric es un proyecto blockchain privado, de código abierto de la Fundación Linux, orientada al uso empresarial, fue concebida para que se adapte a las necesidades de cada organización debido a que, las Blockchain públicas no son elegidas por las empresas que necesitan manejar altos niveles de privacidad, confidencialidad y seguridad en sus transacciones.

Hyperledger fabric deja de lado el anonimato, es decir, que todos los participantes de la red deben ser identificables, adicionalmente, esta blockchain maneja una arquitectura modular y configurable justamente porque al ser su uso empresarial, se requiere que sea personalizable. Lo anterior es posible porque cada nodo de la red, el mecanismo de consenso y los servicios de membresía son plug-and-play, lo que hace que la plataforma pueda alcanzar un alto rendimiento en las transacciones.

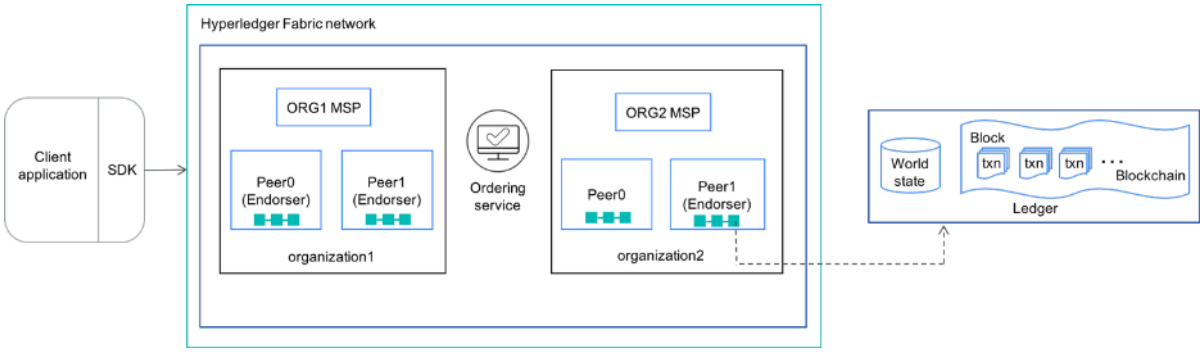


Figura: 12. Componentes de una red Hyperledger Fabric. Fuente: (Maheshwari, 2018)

Una red Hyperledger Fabric tiene estos componentes (figura 12):

- **Activos.** Un activo es cualquier cosa que tenga valor. Un activo tiene estado y propiedad. Los activos se representan en Hyperledger Fabric como una colección de pares *clave-valor*.

- **Libro mayor compartido.** El libro mayor registra el estado y la propiedad de un activo. El libro mayor consta de dos componentes:
  - ✓ *Estado mundial:* describe el estado del libro mayor en un momento dado. Es la base de datos del libro mayor.
  - ✓ *Blockchain:* es un historial de registro de transacciones

- **Smart Contract**. Los contratos inteligentes de Hyperledger Fabric se denominan chaincode. Chaincode es un software que define activos y transacciones relacionadas; en otras palabras, contiene la lógica empresarial del sistema. El chaincode se invoca cuando una aplicación necesita interactuar con el libro mayor y se puede escribir en Golang o en Node.js.

- **Nodos pares.** Los pares son un elemento fundamental de la red porque albergan libros de contabilidad y smart contracts.Un par ejecuta el smartcontract, accede a los datos del libro mayor, respalda transacciones e interactúa con las aplicaciones o respalda otros pares. Cada smart contract puede especificar una política de endoso, que define las condiciones necesarias y suficientes para una transacción válida.

- **Canal.** Los canales son una estructura lógica formada por una colección de pares. Esta capacidad permite a un grupo de pares crear un libro mayor de transacciones por separado.

- **Organizaciones**. La red Hyperledger Fabric se construye a partir de los pares que pertenecen y son aportados por las diferentes organizaciones que son miembros de la red. La red existe porque las organizaciones aportan sus recursos individuales a la red colectiva. Los pares tienen una identidad (certificado digital) asignada por un proveedor de servicios de membresía de su organización propietaria.

- **Proveedor de servicios de membresía (MSP).** El MSP se implementa como una autoridad de certificación para administrar los certificados utilizados para autenticar la identidad y los roles de los miembros. Ninguna identidad desconocida puede realizar transacciones en la red Hyperledger Fabric.

- **Servicio de pedidos.** El servicio de pedidos empaqueta las transacciones en bloques que se entregarán a los pares en un canal. Garantiza la entrega de la transacción en la red.

A diferencia de las implementaciones de blockchain público, Hyperledger Fabric cumple los cuatro elementos clave blockchain para uso empresarial (Gupta, 2017):

- ***Red autorizada:*** membresía definida colectivamente y derechos de acceso dentro de su red empresarial.

- ***Transacciones confidenciales:*** brinda a las empresas la flexibilidad y seguridad para que las transacciones sean visibles para partes seleccionadas con las claves de cifrado correctas.

- ***No depende de las criptomonedas***: no requiere minería y costosos cálculos para asegurar transacciones. No se asigna un costo por transacción.

- **P*rogramable*:** aprovecha la lógica incorporada en Smart contratcs para automatizar los procesos comerciales en la red.

3.4 Quorum

Quorum es un desarrollo de código abierto realizado por JP Morgan, es una bifurcación de Ethereum pensada para el sector empresarial, es decir, es una plataforma que se basa casi totalmente en Ethereum pero que agrega algunas funcionalidades de privacidad y protección de información, transacciones y smart contracts privados, que la hacen más adecuada para ser utilizada entre empresas, además, cuenta con las siguientes particularidades:

- Sus redes son permisionadas (privadas sin opción pública). Por ello, está centrado en la privacidad entre bloques.
- Reducción del coste de transacción y escalabilidad.
- El mecanismo de consenso de Quorum se denomina QuorumChain, que se basa en tiempo y votación por mayoría. También se introduce otra característica llamada Constelación que es un mecanismo de propósito general para enviar información y comunicación cifrada entre compañeros. Además, los permisos a nivel de nodo se rigen por smart contracts.
- Reducción de tiempo de escritura y validación de bloque (prácticamente instantánea).

Para el intercambio del contenido de la transacción, además del componente dedicado a gestionar el blockchain, cada nodo de Quorum utiliza los siguientes componentes(Bashir, 2017b):

- ***Gestor de transacciones (transaction manager).*** Este componente permite el acceso a datos de transacciones cifradas. También gestiona el almacenamiento local y el intercambio de transacciones con los otros nodos involucrados en la transacción.
- ***Enclave criptográfico.*** Es el componente que se encarga de proporcionar servicios criptográficos (encriptar/desencriptar) para garantizar privacidad de la transacción. Adicionalmente, es el responsable de almacenar claves privadas de gestión.
- ***QuorumChain***. Es un mecanismo de consenso tolerante a fallas bizantinas que permite la verificación y circulación de votos a través de transacciones en la red blockchain. En este mecanismo se utiliza un Smart contract para gestionar el proceso de consenso y los nodos pueden recibir derechos de voto sobre qué nuevo bloque debe aceptarse. Una vez que un número apropiado de votos es recibido por los votantes, el bloque se considera válido. Los nodos pueden tener dos roles, a saber, votante o creador.
- ***Gestor de Redes (network manager).*** Este componente proporciona una capa de control de acceso para la red autorizada.

Quorum utiliza una sola Blockchain para almacenar tanto los Smart contracts privados como los públicos, siendo los nodos de la red los encargados de validar las distintas transacciones, utilizando algunos de los componentes mencionados anteriormente. Tanto los smart contracts que sean de tipo privados como las transacciones son expuestas solo a las entidades involucradas. Como se mencionó anteriormente, al ser una plataforma desarrollada como una bifurcación de Ethereum, utiliza la misma tecnología base, de allí que los smart contracts dentro de Quorum son desarrollados utilizando Solidity y se ejecutan sobre la misma máquina virtual (EVM).

3.5 RSK

RSK surge como proyecto en 2015 en Argentina y lanza su Blockchain en producción en enero de 2018. Los fundadores justifican la creación de la plataforma, principalmente para resolver la carencia de implementación de smart contracts en la red de Bitcoin.

RSK es la abreviatura de Rootstock y es la fusión de dos plataformas: QixCoin y Ethereum. QixCoin, nombre inicial que recibió RSK, fue un proyecto de blockchain con criptomoneda propia que introducía el concepto de pago por ejecución, similar al Gas de Ethereum.

Además, mantiene algunos conceptos inherentes a Ethereum como su interfaz, su máquina virtual (VM) y su lenguaje de programación (Solidity) para la creación de smart contracts. De lo anterior se colige que estas dos plataformas son altamente compatibles.
RSK es una Bitcoin-sidechain, es decir, RSK tiene su propia Blockchain pero no su propia criptomoneda.

RSK utiliza el mecanismo de consenso PoW para validar las transacciones y añadir bloques a su cadena, con un adicional, soporta *minería fusionada* (merge-mining), esto es que el trabajo que realizan los mineros para resolver el acertijo y ganar bitcoins, también sirve para el minado sobre la Blockchain de RSK, sin requerir equipos extra ni pérdida de poder de cómputo.

En RSK se agregan al menos tres factores que no están en la red Bitcoin: *la Federación RSK, la minería fusionada y DECOR+.*

- ***Federación RSK**:* un grupo de 15 miembros en un sistema semi-intermediado, los cuales se encargan de congelar y descongelar los BTC según sea necesario. Esos 15 miembros son mayormente compañías destacadas del ecosistema con capacidad técnica para mantener y administrar su propio nodo. A cambio de sus servicios, la Federación recibe el 1% de las comisiones por transacción generadas en RSK. Por otra parte, si se diera el caso de que se quiera cambiar a algún o algunos miembros de la Federación, cada uno de los miembros actuales deberá votar por medio de un Smart contract público para aceptar o rechazar a los nuevos miembros.

- ***Minería fusionada (merge-mining):*** básicamente, se trata de minar dos o más criptomonedas distintas al mismo tiempo**,** con las soluciones de minería de una sola de ellas, los mismos equipos y casi el mismo software. Sin embargo, las criptomonedas que se desean minar deben tener el mismo algoritmo y una de ellas debe tener su código ya modificado para permitir la merge-mining.

- ***DECOR+:*** Es el acrónimo de Deterministic Conflict Resolution (Resolución de conflicto determinista)**.** Se trata de un protocolo cuyo principal objetivo es que los mineros no realicen *minería egoísta* y haya una justa distribución de recompensas. Esta *minería egoísta* se da cuando un grupo de mineros confabulan entre ellos para reunir mayor poder de minado que los demás, minar una cadena más larga y secreta, y añadirla más tarde a la blockchain pública, reemplazando así a los bloques nuevos añadidos por otros mineros con menor capacidad y arrebatándoles su recompensa. DECOR+ previene este tipo de comportamiento en la cadena de RSK. Asimismo, DECOR+ permite la creación de una política de incentivos por parte de todos los mineros para asignar recompensas a los bloques dependiendo de las circunstancias.

Una vez descritas algunas de las plataformas para el desarrollo de smarts contracts (se seleccionaron como plataformas de comparación algunas que tienen influencia en la capitalización del mercado mundial, gran potencial para implementar soluciones para diversos sectores de bienes y servicios; y que además, cuentan con posibilidad de integración de Smart contracts), merece la pena indicar que no se puede afirmar que una sea mejor que la otra, sin embargo, para tomar la decisión de implementar alguna de ellas, dependerá de aspectos técnicos como los requerimientos y lenguajes de programación utilizados en cada caso y de aspectos ideológicos, por ejemplo, mayor o menor afinidad por ciertos protocolos de consenso y mecanismos de gobierno más o menos descentralizados.

Ahora bien, para el desarrollo de las aplicaciones de las que trata esta tesis, se requiere de una plataforma de desarrollo que permita la incorporación de Blockchain y smart contracts; dentro de las cuales, se identifican varias herramientas disponibles para este tipo de desarrollos. La tabla 6 presenta un cuadro comparativo que evalúa este tipo de plataformas bajo los siguientes criterios: tipo de blockchain, consenso, criptomoneda, lenguaje de programación de smart contracts y modificación de smart contracts".

- **Mecanismo de consenso:** Este criterio es importante porque los mecanismos de consenso determinan atributos característicos como: escalabilidad, grado de descentralización, seguridad del sistema mediante la tolerancia a fallos y consistencia del ledger.

- **Lenguajes permitidos:** Es relevante comparar cuáles son los lenguajes permitidos en cada plataforma, ya que permiten obtener indicadores sobre cuántos son los desarrolladores que los usan, la documentación que exista al respecto y las características específicas de cada lenguaje.

- **Tipo de blockchain:** Es importante saber a qué tipo de blockchain pertenecen las plataformas objeto de estudio, sean estas: públicas, privadas o permisionadas. Las principales diferencias entre estos tipos de Blockchain son la forma en la que se comparte el ledger y el permiso para participar en el sistema.

- **Modificación de smart contract desplegado:** Se analiza si la plataforma brinda la posibilidad de actualizar un smart contract una vez desplegado. Se esperaría que la modificación de un smartcontract puede mejorar la calidad general, ya que permite corregir errores o introducir mejoras en contratos desplegados. Sin embargo, al brindar esta posibilidad, se agrega más vulnerabilidad al sistema porque este recurso puede ser usado de forma maliciosa.

Aunque puede resultar contradictorio hablar de smart contracts modificables en el contexto de la inmutabilidad que ofrece la tecnología Blockchain, sin embargo, en la mayoría de las plataformas estudiadas que permiten esta modificación, no modifican el código ya desplegado, sino que implementan algún patrón de diseño específico para cada lenguaje o plataforma dentro de sus smart contracts. En algunos casos las plataformas también implementan operaciones que permiten deshabilitar ciertos contratos. Es decir, el código del contrato sigue estando almacenado dentro de la Blockchain, pero el mismo no puede ser consumido por ningún cliente.

Plataforma	BlockChain	Consenso	Criptoactivo	Modificar Smart contracts	Lenguaje Smart contracts
Ethereum	Público	PoW en transición a PoS	Ether (ETH)	Si	Solidity, LLL, Serpent
Hyperledger Fabric	Privado	PBTF/SIEVE	No tiene	No	Node js -Go -Java
Lisk	Público y Permisionado	DPoS	LSK	No	Node.JS Javascript
Quorum	Permisionado	Quorum chain basado en voto	ETH	Si	Solidity
TRON	Público	DPoS	TRX	Si	Solidity
EOS	Público	DPoS	EOS	No	C++
RSK	Público	PoW (Decor+)	No tiene	Si	Solidity

Tabla 4: Comparativo plataformas blockchain con smart contracts. Elaboración propia a partir de: (Rodríguez Molano et al., 2020)

En primer lugar, se descartaron las plataformas que no integran Smart contracts, por ello, aunque sean reconocidas y ocupen lugares importantes en el mercado de capitalización mundial, no aparecen en la tabla comparativa. Por otro lado, de las plataformas que se presentan en la tabla, para resolver la problemática de esta tesis, se descartan Tron, Hyperledger Fabric y Quorum, debido a que se limitan a funcionar como gestores de recursos financieros y las dos últimas, además funcionan bajo blockchain privado o permisionado.

Las restantes plataformas cumplen los criterios establecidos, siendo Ethereum pionero en Smart contracts e implementación en diversas áreas y RSK, que habilitan la creación e

implementación de aplicaciones descentralizadas, serían las plataformas recomendadas para adelantar esta solución.

Adicionalmente, se determinan como posibles aplicaciones Ethereum y RSK, debido a que se soportan en tecnología Blockchain pública, es decir que permite acceso sin restricción y prevalece el parámetro de transparencia (Bashir, 2017a), lo que permite que la participación sea abierta, sin que se pierdan los atributos de seguridad y transparencia; esto es justo lo que se requiere para el sector público.

4. Campos de aplicación de blockchain con smart contracts

Gracias a la automatización y a la desintermediación en los procesos que brindan los Smart contracts en conjunto con la blockchain sugieren las utilidades de uso a nivel empresarial, ya que podrían incrementar la eficiencia y la velocidad de operación, así como la reducción de costes. A continuación, se presenta una referencia no exhaustiva de sectores que han encontrado o tienen potencial de soluciones a partir de la de aplicación de blockchain con smart contracts:

Sector	Descripción -caso
Gobierno y administraciones públicas	Seguridad digital y todo lo relacionado con la información que manejan las administraciones públicas y que requiere la ejecución de un evento, ofreciendo la posibilidad de un ahorro importantísimo en tiempo, dinero y estructuras. Por otro lado, a la hora de legislar, tener visibilidad de todos los tipos de contratos y acuerdos firmados y cómo se ejecutan, permitirá también a los reguladores aprender sobre lo que está pasando, qué necesidades van surgiendo y tomar decisiones basadas en la realidad.
Automovilístico	El automovilístico será uno de los mercados que más necesidad de servicios pueda desarrollar, por ejemplo, para pagar parkings, peajes, impuestos, seguros, y también en caso de accidentes para poder determinar causas y responsabilidades.
Compañías de seguros	Todo tipo de pólizas que necesitan renovaciones, comunicaciones, cancelaciones, reclamaciones, ejecuciones.
Banca	Todo lo que se refiere al sector bancario está recogido en contratos que en muchas ocasiones se podrán automatizar, así como pagos, transferencias de importes, vencimientos, etc.
Salud	No sólo para almacenar los historiales de los pacientes de forma codificada y segura permitiendo el acceso sólo a personas o entidades autorizadas, sino para autorizar de forma automática pruebas o intervenciones que cubran ciertas pólizas, etc.
Mercado inmobiliario	Tanto para la compra de viviendas como para controlar el mercado de los alquileres.
Cadena de suministro	• Según Jeff Garzik, se pueden ejecutar smart contracts que indiquen: 'Si recibo un efectivo a la entrega en algún lugar en un mercado emergente en desarrollo, entonces este otro [producto], muchas conexiones en la cadena de suministro, provocará que el proveedor cree un nuevo registro como que ese producto acaba de ser entregado en ese mercado en desarrollo'. • Vehículos autónomos que hacen entregas P2P. • Seguimiento de piezas desde su creación hasta su ensamblaje final (lo hace la blockchain). El reto se encuentra en que mediante smart contracts se pueda además hacer un seguimiento de procesos y de procedencia de metadatos históricos acerca del elemento de manera automatizada en un registro de datos distribuido, para tener información confiable que permita la mejor detección de errores y oportuna toma de decisiones al mismo tiempo que se reemplazan los contratos hechos con papel

Auditorías	Al tener smart contracts autoejecutables y validables por un ordenador, los auditores pueden observar y realizar auditorías practicamente en tiempo real.
Conciliaciones	Resolución de disputas o conflictos entre personas donde los miembros de la misma comunidad intervienen como jurados
Casas de cambio	Intercambio de criptomonedas y compra de las mismas con dinero fiat o incluso, con dólares.
Marketplace/compras/ventas	Compras con pagos directos por medio de la red de bitcoin y utiliza esta criptomoneda como medio de intercambio. Orientado al sector laboral donde freelancers y clientes se encuentran, se comunican y facilitan sus pagos de manera privada y segura.
Votaciones	Para garantizar la privacidad del voto, tanto durante su recolección como acciones que se desencadenarían correspondiente a los resultados obtenidos.
Contratos legales, comerciales, de arrendamiento	Contratos hechos sobre papel son smart contracts que ayudarían a reducir documentación compleja para el seguimiento de los elementos, envío y sus movimientos.
Préstamos	Podrían almacenarse como smart contracts en el blockchain, junto con la información de las garantías de la propiedad. Si el deudor no efectúa un pago, el smart contract podría revocar automáticamente las claves digitales que le dan acceso a las garantías.
Dividendos, inversiones	Pagos automáticos a stake holders de empresas, regalías a artistas y otro tipo de contribuciones.
Herencias	Podrían automatizarse estableciendo la asignación de activos tras el fallecimiento. Una vez que el smart contract puede verificar la condición de activación, en este caso el fallecimiento, el contrato entra en vigor y los activos se reparten.
Mercados capitales	Los valores basados en pagos y derechos que se ejecutan según unas reglas predefinidas se pueden escribir como smart contracts. Hay experimentos para la emisión y monitoreo de bonos inteligentes y para la gestión de mercados de valores privados.
Monederos de criptomonedas	• Los monederos controlados por smart contracts podrían incluir muchos tipos diferentes de controles complejos, desde límites de reintegro diarios hasta la concesión o la rescisión del acceso a entidades específicas. • Dinero programable, que puede establecerse de modo que se gaste únicamente en determinados tipos de activos, en una zona geográfica, entre dos fechas, etc.

Tabla 5: Resumen campos de aplicación de blockchain con smart contracts

Por último, en el mundo de la dirección y organización de empresas en general se abre un abanico casi infinito de posibilidades que cambiará la forma en la que se realizan labores cotidianas tal y cómo las conocemos hoy en día.

CAPÍTULO 5. APLICACIONES DE BLOCKCHAIN Y SMART CONTRACTS EN EL SECTOR PÚBLICO

La administración pública es uno de los sectores de mayor relevancia e incidencia debido a que es el responsable de establecer y garantizar la implementación de políticas públicas que favorezcan el bienestar social y económico de los ciudadanos. No obstante, la actual no se ha caracterizado por brindar respuestas integrales a las demandas de la sociedad, al contrario, es percibida como lenta y burocrática, su ineficiencia ha ocasionado que los ciudadanos pierdan la confianza y credibilidad en estas instituciones.

Se trata entonces de recuperar la imagen, la confianza y la credibilidad con modelos más transparentes, rápidos e integrados a la vida diaria de los ciudadanos, que además permita su participación e incidencia y en este sentido, el sistema blockchain no solo sirve para empresas que buscan beneficios y rentabilidad, sino también para entidades públicas, tanto en el ámbito de ***gobierno, educación*** y ***sanidad***; como en ***redes energéticas, sistemas de transporte*** y ***servicios sociales***, entre otros (Antonio et al., 2018) (Tapscott, Don, Tapscott, Alex, 2019).

Teniendo en cuenta que la corrupción y la falta de transparencia constituyen las mayores problemáticas en la actual administración pública y que pese a que es crucial cambiar el comportamiento de una institución, no es posible obligar al ser humano a optar por unas conductas, y menos aún, desde el alcance de esta investigación, no obstante, a partir de la tecnología **sí** se pueden limitar las decisiones y acciones que se pretenden materializar y que vayan en contra de una correcta administración pública.

En este sentido, trascendiendo el plano económico, se pueden manejar los *smart contracts* como pacto de condiciones en cualquier ámbito, que no requieren de intermediarios que validen o que vigilen su cumplimiento, dado que no da lugar a interpretaciones equívocas (Moralejo, 2018), debido a que se ejecutan a sí mismos y se almacenan en una blockchain, que nadie puede controlar y en los que todos pueden confiar.

Así las cosas, el sistema blockchain permitirá automatizar estos procesos y garantizará la integridad de sus transacciones, concesiones administrativas, registros e importantes decisiones, por lo que los funcionarios no podrían ocultar pagos ni registros oficiales o de otras manipulaciones desde dentro o desde afuera y propiciará un mayor control, trazabilidad y transparencia en los procesos(Triana Casallas et al., 2020).

No es desconocido que actualmente, la mayor parte de administraciones públicas se encuentran pasando por una crisis de credibilidad en muchos campos, provocada en parte, por la carencia de procesos innovadores y por las inadecuadas formas (estrategias) de participación ciudadana en la veeduría de la gestión y para la toma de decisiones.

La búsqueda de recursos innovadores para repensar la comunicación y relación con los ciudadanos debe ser, por tanto, una constante en la gestión de las organizaciones y las ciudades. La aparición de nuevas corrientes en materia tecnológica, como las

ciudades Inteligentes o internet de las cosas, no pueden dejar de lado la relación con el ciudadano y la forma en la que éste se relaciona y toma parte en la propia gestión del territorio (Moralejo, 2018).

Aunque durante los últimos años, términos como gobierno abierto, transparencia y administración electrónica, son invocados como si se tratara de la modernización de la administración pública, esta sigue adoleciendo de capacidad para incorporar satisfactoriamente aquellos instrumentos que permitirían un funcionamiento de la actividad pública más eficaz y eficiente, reduciendo el ejercicio a poner a disposición de los ciudadanos, la información y los datos relacionados con su gestión.

Por ello, que deba considerarse una administración con apertura a la adaptación y transición tecnológica en la que la blockchain, posibilitando resolver las demandas anteriormente indicadas, al propiciar que los ciudadanos, las empresas y organizaciones de la sociedad civil, puedan no solamente acceder a información relevante, sino también, mejorar los servicios públicos y participar en la toma de decisiones de manera más activa e incidente.

Concretamente, como garantía en materia de ***transparencia***, "la tecnología blockchain permitiría contar con sistemas de registro que facilite la consulta y el seguimiento de las operaciones del Estado, favoreciendo un cambio en el ejercicio de la práctica institucional que promueva una mayor apertura a la información, permitiendo la generación de controles ciudadanos y facilitando una mayor eficiencia en la gestión de los entes de control.

A través de dichos registros, se creará una identidad digital propia de cada elemento u operación que permitirá conocer su historia y realizar su seguimiento en función de los niveles de transparencia que se establezcan y los permisos que se otorguen"(Lucas, 2019).

En la tabla que sigue, se presenta de manera resumida, las aplicaciones de blockchain y Smart contracts en el sector público.

País	*Aplicación*
Estonia	Voto online
Australia	Voto electrónico
Reino Unido	Voto online
Suecia	Registro de títulos de propiedad
Georgia	Registro de títulos de propiedad
Honduras	Registro de títulos de propiedad
Ghana	Registro de títulos de propiedad
Rusia	Registro de títulos de propiedad
Suiza	Voto online
Dinamarca	Voto online
Francia	Voto online
Holanda	Voto online
Australia	Identidad digital
China	Administración de impuestos y emisión de facturas electrónicas
Dubai	Verificación de registros médicos electrónicos entre hospitales y clínicas
Italia	Identidad digital

Vancouver	Repositorio público de reclamos verificables sobre organizaciones.
Vancouver	Registro de tierras
Estados Unidos	Voto electrónico
Estados Unidos	Eliminación de registros de papel
Estados Unidos	Administrar la identificación de los residentes del estado, así como tokenizar los activos en el sector público para mejorar la eficiencia y reducir el fraude de derechos.
Estados Unidos	Transferencia de la propiedad
	Identidad digital
Estados Unidos	Urbanismo y espacio público (planificación y diseño de ciudades
	Seguridad Ciudadana
Argentina	Identidad digital para mejorar el acceso de los ciudadanos a los servicios gubernamentales
Malta	Programa de credenciales basado en blockchain que verifica instantáneamente las credenciales académicas
Ucrania	Subastas de prueba
Sierra Leona	Historial crediticio
Tanzania	Auditoría de nóminas públicas

Tabla 6: Aplicaciones de blockchain y smart contracts en el sector público. Elaboración propia a partir de: (Rodríguez Molano et al., 2020)

En seguida se presentarán con algo más de detalle algunas de las proyecciones de blockchain para categorías o casos de uso específicos:

- **Registro de títulos de propiedad:** Los proyectos relacionados con esta temática pretenden mitigar la posibilidad de corrupción, al tiempo que proporciona a los clientes recibos electrónicos seguros y verificables. Permite auditorías independientes de contratos inteligentes, así como la gestión descentralizada de la identidad, tiene el potencial de simplificar el proceso de registro público y el mantenimiento continuo a través de canales digitales.
  Blockchain permite la "tokenización", de los activos, incluso los inmuebles, de tal manera que pueda pueda realizarse su transmisión con la confianza de que el vendedor es quien dice ser y puede responder del pago, el comprador es quien dice ser y es el dueño del inmueble que se transmite y un Smart contract se encarga de verificar automáticamente todas esas circunstancias y realizar el pago del inmueble y el registro del mismo a nombre del nuevo propietario(Triana Casallas et al., 2020).
- **Identidad digital:** Los registros de nacimiento le permiten al estado emitir una identidad digital vinculada al nacimiento de una persona que se podría administrar en un ledger, agregando atributos a medida que el ciudadano interactúa con diferentes agencias a lo largo de su vida y permitiría a las personas verificar su identidad en solo unos minutos a través de un teléfono inteligente utilizando datos biométricos.
  Bitnation es un proyecto de innovación basado en los smart contracts y la tecnología de Ethereum, y se define como una "Nación Voluntaria Descentralizada sin Fronteras" o nación digital. Es un proyecto de gobierno abierto, de gobernanza, que propone soluciones para tener documentación de identidad protegida pero demostrable, sistemas de cobertura o de seguros "pública" y generación de trámites, como certificados de nacimiento, entre otros.
- **Administración de impuestos y emisión de facturas electrónicas:** A partir del uso de la blockchain para organizar y administrar el sistema de recaudación de los impuestos y la emisión de facturas electrónicas, con el propósito de disminuir los

niveles de evasión.

- **Verificación de registros médicos electrónicos entre hospitales y clínicas:** El esquema piloto incluirá el proceso manual existente que se reemplazará con un sistema que utiliza la tecnología blockchain "que puede transferir y verificar automáticamente los registros de pacientes.
- **Administrar la identificación de los residentes del estado**, **así como tokenizar los activos en el sector público.** El uso de una plataforma basada en blockchain permitiría a los ciudadanos estatales acceder y almacenar toda su información de identificación, como impuestos, votación y licencia de conducir, entre otros, como nodos descentralizados, con el propósito de mejorar la eficiencia y reducir el fraude de derechos.
- ***Voto electrónico***, descentraliza la responsabilidad y la dispersa entre los nodos participantes, que son los que logran el consenso sobre los datos albergados en la base de datos que al estar basadas en sistemas centralizados y gobernados por una única fuente no garantizan la inalterabilidad de la intención del voto, por lo que cualquier manipulación en la base de datos podría suponer un cambio drástico en los resultados de una votación online; lo que permite dar respuesta a una de las mayores deficiencias de las plataformas actuales.
- ***Gobierno electrónico*** basado en la interconectividad y la descentralización, apertura y seguridad cibernética.

  D-Cent y DECODE, son dos proyectos europeos que pretenden aplicar blockchain a la resolución de problemáticas asociadas a los gobiernos abiertos. El primero de ellos, *D-Cent,* es un proyecto para generar “Tecnologías de Participación Ciudadana” de propiedad pública, pero buscando mayor agilidad e innovación pública. Una de las bases tecnológicas del proyecto es el Blockchain, y se busca aplicarla como solución a temas de gestión democrática de los datos generados por ciudadanos y ciudades (big data), para proteger y asegurar la privacidad y protección de datos con las normativas, o gestionar espacios de debate y deliberación públicos digitales.

  El segundo proyecto *(DECODE)* trabaja sobre la idea de cómo los ciudadanos podrán decidir qué hacer y cómo gestionar sus datos en un escenario de mayor transparencia, automatización y digitalización de los datos de las ciudades e identidades, así como el impacto económico que podrían generar en estas ciudades. Gestión de la Democracia Abierta, Internet de las Cosas y economías colaborativas.

CAPÍTULO 6. POTENCIAL DEL BLOCKCHAIN EN LA CONTRATACIÓN PÚBLICA

El World Economic Forum calcula que los costos generados por la corrupción ascienden a más de 2,6 trillones de dólares (más del 5% del Producto Bruto Mundial-PBM) mientras que en un informe titulado *Myths and realities of governance and corruption*, el Banco Mundial ha estimado que se paga únicamente en concepto de coimas y sobornos más de un trillón de dólares por año.

Esto indica que cerca del 2% del PBM termina en manos de agentes de gobierno corruptos que intervienen en la ejecución de diversos actos de los Estados y que, de implementarse eficientemente la tecnología blockchain con smart contracts, podrían quedar sin el poder que utilizan con fines ilegítimos en beneficio propio o de terceros, por ejemplo en licitaciones públicas y otros tipos de contratación, cuyos procesos sean asignados de forma automática a las empresas que hayan sido los mejores oferentes o las mejores candidatas y que no tengan forma de direccionarlas a aquellas cuyos funcionarios hayan ofrecido o recibido algún tipo de incentivo extraoficial o dádivas, eliminando la posibilidad de que exista un intermediario que pueda facilitar el perfeccionamiento de ese pago o que el contratista reciba cuantías sin haber ejecutado la obra pactada.

Utilizando tecnología blockchain, cada una de las transacciones pueden ser trazadas hasta su origen lo que aporta significativamente a la persecución de un eventual acto de corrupción. El diferencial que aportan los smart contracts sobre otras soluciones basadas en la tecnología blockchain está en la auto ejecutoriedad de las instrucciones y de las operaciones que regulan, lo que genera una imposibilidad o, al menos, un incremento en la dificultad de ejecutar actos de corrupción.

Con base en lo anterior, es importante enunciar el tema de la contratación pública como uno de los mayores generadores de corrupción, dado que representa una parte sustancial del dinero de los contribuyentes a nivel mundial, y sigue siendo la actividad más vulnerable al despilfarro, el fraude y la corrupción. Evidencia de ello, la constituyen los datos que pone de manifiesto la Alianza para el Gobierno Abierto (Open Government Partnership) que supone alrededor del 50% del gasto total de un gobierno típico en países de ingresos bajos y medianos, y cerca del 30% en países de altos ingresos (Open Government Partnership, 2019).

En promedio, entre el 10% y el 20% de los presupuestos de adquisición pueden desperdiciarse en función del grado de corrupción, despilfarro e ineficiencias. La corrupción distorsiona un sistema de adjudicación justo, limita la igualdad de oportunidades entre licitadores, perjudicando la competencia y en consecuencia, disminuyendo la calidad de las obras, suministros y servicios públicos, lo que acaba también socavando la confianza en las instituciones públicas (Open Government Partnership, 2019).

Recientemente la Comunicación COM (2017) 572 final de la Comisión al Parlamento Europeo, al Consejo, al Comité Económico y Social Europeo y al Comité de las Regiones, reconoce de manera explícita, que las disposiciones más estrictas en materia de integridad

y transparencia de las directivas tienen como finalidad la lucha contra la corrupción y el fraude, y presenta una estrategia de contratación pública que establece el marco político general y define prioridades claras para mejorar la contratación en la práctica y apoyar las inversiones en el seno de la UE, en la que la lucha contra la corrupción en la contratación pública ocupa un lugar prevalente (Comité et al., 2018).

Se debe agregar que, la Organización para la Cooperación y el Desarrollo Económicos (OCDE) en sus "Principios para la integridad en la Contratación Pública" (Organisation for Economic Co-operation and Development (OECD), 2017) sostiene que la integridad en la contratación pública se implementa en la práctica a través de cuatro principios: la transparencia, la buena gestión, la prevención de la mala conducta y la rendición de cuentas.

Por otro lado, el Informe global Open Government Partnership (OGP), destaca que "la corrupción en la procuración pública puede reducir el valor de los contratos hasta en un 15% (dependiendo de la estimación utilizada). La contratación abierta – incluyendo la publicación de contratos y la participación, monitoreo y supervisión ciudadana – han demostrado que tiene el potencial de generar ahorros fiscales, reducir la corrupción y fortalecer la participación de las empresas"(Open Government Partnership, 2019), con lo que se demuestra el gran potencial de block chain para mejorar los indicadores de lucha contra la corrupción a nivel mundial.

En concordancia con lo expuesto anteriormente, la simple digitalización de los procedimientos de contratación de principio a fin, y el establecimiento de publicación de prácticamente todos los eventos del procedimiento a través de las aplicaciones en formatos abiertos e interoperables, coherentes con la legislación de cada país, y la multiplicación de los mecanismos de control, introducen en el proceso de contratación, elementos de transparencia que por sí mismos reducen las posibilidades de fraude, corrupción e ineficiencia.

Sin embargo, las características de inmutabilidad, confidencialidad, trazabilidad y transparencia de Blockchain junto con la automatización y desintermediación que implican los Smart Contracts, la hace especialmente útil en la lucha contra la corrupción y el fraude.

En complemento, Botto y Castrovinci han apuntado como novedades que reportaría el uso de la Blockchain, además de la posibilidad de establecer un mecanismo de control de la integridad de la documentación y el proceso realizado por las propias empresas licitadoras, la reducción en el calendario de los procedimientos asociados a las licitaciones, dada justamente porque un Smart contract ya ha definido las reglas de ejecución. (*La Blockchain Negli Appalti Pubblici, Come Utilizzarla: I Vantaggi | Agenda Digitale (Italian)*, n.d.).

De hecho, si se piensa ir más allá de la adjudicación, Morris Gitonga señala que el uso de la tecnología Blockchain puede prevenir la corrupción en la gestión de las licitaciones adjudicadas en la medida que todos los sucesos son transparentes y verificable por cada licitador (*Using Blockchain Technology to Eliminate Corruption in Developing Nations - Coinweez*, n.d.).

En conclusión, dadas las características de inmutabilidad, confidencialidad, trazabilidad y transparencia de Blockchain junto con la automatización que implican los Smart Contracts, hace de los procedimientos de contratación en el sector público, el campo ideal para su implementación, lo que no eliminaría la corrupción, pero si permitiría su detección temprana para que se puedan tomar medidas correctivas y preventivas.

Los elementos que conforman las aplicaciones seleccionadas, son construidos desde la estructura de datos BlockChain pública y los Smart contracts que de allí se derivan, aportan a la herramienta versatilidad sin dejar de lado la seguridad, configurando un espacio propicio para ser un eje de interacción, entre el estado, los licitadores y la ciudadanía interesada en el proceso de contratación, donde cada uno puede utilizar y participar en la red, de esta forma se conecta eficientemente a los diversos actores presentes, agilizando las etapas del proceso, teniendo un mejor acceso y control de los datos, pero permitiendo la transparencia en la evaluación y adjudicación.

Experiencias del uso de Blockchain en contratación pública

Existen diversas iniciativas para el uso de Blockchain en la contratación pública que a continuación se presentan, sin embargo, no se encontró información de proyectos en funcionamiento salvo el caso de Perú:

- En ***Perú,*** el organismo gubernamental de compras públicas "Perú Compras", incluyó el uso de Blockchain en abril de 2018, para registrar las órdenes de compras de forma digital. Desde entonces, el país ha registrado cerca de 50.000 órdenes de compra a través de su plataforma de Catálogos Electrónicos. Perú reconoce la aplicación de tecnología Blockchain como una herramienta eficaz para brindar transparencia al ámbito de las contrataciones públicas. Perú Compras opera a través de la red Blockchain LAC-Chain, un proyecto descentralizado del Banco Interamericano de Desarrollo (BID).
- En ***México***, asistentes a Talent Land 2018 pudieron apreciar cómo una unidad compradora puede realizar una convocatoria de licitación y cómo una empresa puede postularse para ofrecer sus productos y servicios al gobierno, todo a través del blockchain, lo que hace que las transacciones sean inmutables y completamente trazables. Este proyecto actualmente cuenta con un diseño, un prototipo en versión alfa con transaccionalidad, se espera dejar que cumpla un periodo de madurez hasta alcanzar una versión beta y entonces evaluar la posibilidad de aplicarlo en un caso real que vaya a la par de Compranet, el sistema transaccional que permite a las instituciones públicas de México realizar procedimientos de contratación de manera electrónica, mixta o presencial. El proyecto no se ha ejecutado debido a problemas de regulación que surgen al introducir una tecnología en un proceso administrativo del gobierno.
- En 2018, ***Canadá*** realizó con éxito la primera prueba de uso la tecnología pública de Blockchain (en Ethereum) en contratos públicos con el fin de permitir una administración transparente de los contratos del gobierno. A la fecha no se conocen resultados o indicadores que demuestren que blockchain en la contratación ha reducido los niveles de corrupción o incrementado los de transparencia.
- En ***Estados Unidos***, la Agencia gubernamental de Servicios Generales (GSA) de Estados Unidos, a través de su Oficina de Tecnología Ciudadana Emergente, anunció el lanzamiento del Programa Federal de Blockchain de Estados Unidos" con el

objetivo de que las agencias federales y empresas estadounidenses puedan explorar la tecnología Blockchain.

- En ***Chile***, en julio de 2018, inicia prueba piloto para el uso de Blockchain en la contratación pública que les permita integrarla a la plataforma de compra pública "Chile Compra", con lo que se busca mejorar los índices de confianza, transparencia y menor burocracia.
- En ***Japón*** también se encuentra en pruebas para la aplicación del Blockchain a la Contratación Pública. Finalizando 2019 se sostuvo una reunión entre Perú Compras y la entidad coreana de contratación, en la que los representantes asiáticos se mostraron interesados en conocer más sobre la aplicación de la tecnología en el sector de contrataciones públicas.

Blockchain en el procedimiento de licitación y adjudicación

El uso de blockchain en contratación pública debe orientarse a responder a distintos modelos según la legislación del país donde se implemente, pero existen temas y actividades comunes que han sido identificadas en el modelo tipo para presentación y evaluación de ofertas propuesto por Freya Sheer Hardwick, Raja Naeem Akram, y Konstantinos Markantonakis (Hardwick et al., 2018), que se presenta en la figura 13:

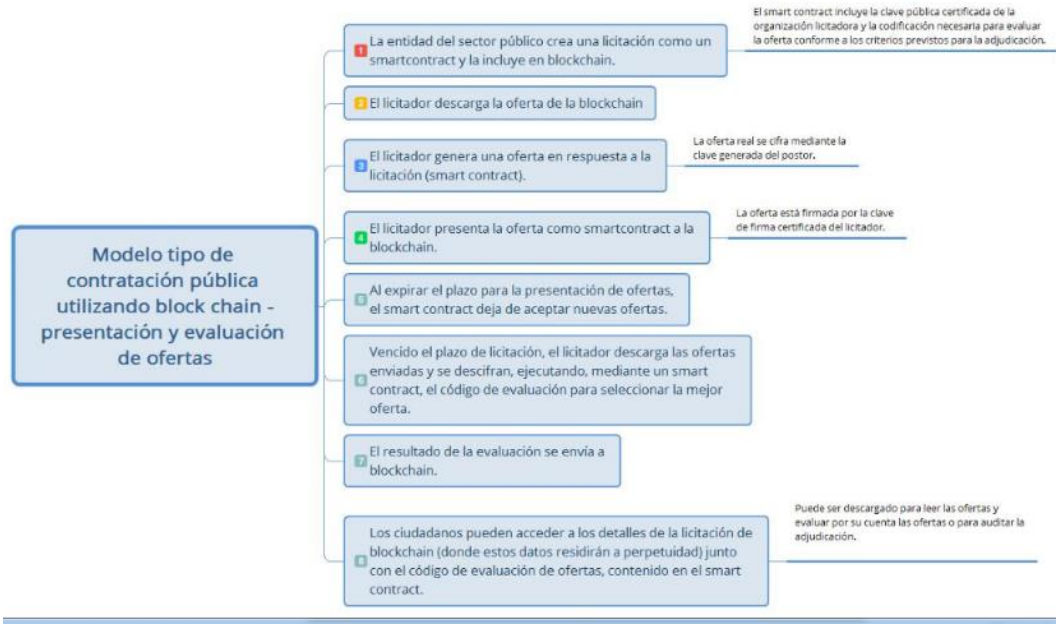


Figura: 13. Ejemplo de evaluación de ofertas bajo enfoque smartcontract (basado en (hardwick et al., 2018))

Blockchain en la confidencialidad de los documentos del proceso contractual

La tecnología de Blockchain puede actuar como garantía de seguridad y confidencialidad en relación con la información que en el proceso de licitación las entidades del sector público ponen en manos de los licitadores, en cuyo caso la prueba de la existencia del consentimiento para su acceso se marca y almacena en Blockchain.

La obtención del consentimiento debe ser un "bloqueo" antes del acceso a información clasificada como confidencial por cualquier sujeto ajeno al órgano de contratación. De hecho, con la ayuda de claves criptográficas de Blockchain, nadie podrá acceder a información confidencial hasta que obtenga el consentimiento del titular de dicha información. Cada transacción de blockchain puede tener un bloqueo asociado y

las transacciones pueden estar pendientes y activarse a un tiempo de contrato acordado.

Freya Sheer Hardwick, Raja Naeem Akram, y Konstantinos Markantonakis, citados anteriormente, señalan una serie de requisitos de confidencialidad y seguridad que en cualquier caso debería cumplir un sistema de licitación como el descrito:

- Los licitadores una vez ha subido su oferta a la blockchain no pueden modificarla.
- La organización licitadora no puede leer la oferta hasta que venza el plazo.
- Los licitadores no pueden cambiar las ofertas de otra organización.
- Los Licitadores no pueden ver quién más hizo una oferta.
- Los mineros de la red Blockchain no pueden afectar el proceso de licitación.
- Para garantizar la integridad del proceso sería conveniente que ni siquiera la entidad del sector público pudiera conocer el número de licitadores que han presentado ofertas.

Por lo anterior, la naturaleza descentralizada, transparente y segura del protocolo Blockchain, puede cumplir las condiciones de las entidades públicas respecto a la información confidencial que manejan, obteniendo así un proceso más transparente y confiable para su tratamiento.

Blockchain en el procedimiento de obtención de garantías

La presentación y devolución de las garantías, que no son aplicables a todos los procesos de contratación pública, refieren un novel adicional de complejidad que también pueden ser simplificados y automatizados con la aplicación de blockchain. A continuación, se presenta la secuencia tipo (ver figura 14):

Figura: 14. Ejemplo obtención de garantías bajo enfoque blockchain (basado en (hardwick et al., 2018)

CAPÍTULO 7. MODELOS Y META-MODELOS

La evolución de la ingeniería del software puede describirse como un continuo ascenso en el nivel de abstracción y para ello han sido utilizados los modelos, con mayor ahínco en tiempos recientes (Jesús García Molina et al., 2014). Sin embargo, es importante destacar que el centro de atención por los modelos se da inclusive en procesos donde no se contemple necesariamente la generación automática de código como finalidad, sino que los modelos también tienen un papel fundamental para ayudar a razonar sobre sistemas en desarrollo, para facilitar la comunicación entre los distintos participantes, y para documentar las decisiones de diseño (Bézivin, 2005b).

En las líneas que siguen se presentan las definiciones y generalidades relacionadas con los modelos y meta-modelos.

1. Modelo

Un modelo es una abstracción o representación simplificada de una realidad o sistema (Rumbaugh, 2005) o parte de él, el cual se simboliza a través de un conjunto de artefactos, de manera gráfica o textual (Miller et al., 2003) que describen un aspecto o una parte del modelo y permite razonar sobre ese sistema (realidad), eliminando detalles irrelevantes, concentrándose en aquellos que resultan esenciales. Cada modelo describe un único aspecto particular del sistema, con un propósito específico y es descrito con el nivel de abstracción adecuado para el problema que se ha modelado.

Por su parte, OMG (2019) plantea que un modelo es una representación selectiva de algún sistema cuya forma y contenido se eligen en función de un conjunto específico de inquietudes y está relacionado con el sistema mediante un mapeo explícito o implícito.

Como se mencionó, en tanto que los modelos son abstracciones o simplificaciones de la realidad, pretenden ayudar a gestionar la complejidad que revisten los productos y procesos de desarrollo de software. Los modelos están más cerca del entendimiento humano que el código, de modo que trabajar con modelos será menos propenso al error que trabajar con lenguajes de programación (Jesús García Molina et al., 2014).

En ingeniería se usan los modelos como elementos esenciales para comprender sistemas complejos del mundo real, por ejemplo, para predecir las cualidades del sistema, para razonar sobre propiedades específicas cuando varios aspectos del sistema cambian, o para poder comunicar las características clave del sistema a las diferentes partes interesadas (Atkinson & Kühne, 2007).

Es importante subrayar que de un mismo sistema se pueden tener diversos modelos que, en función de la información relevante que se quiera destacar, pueden ser alternativos y complementarios, por ello, un modelo debe cumplir con las siguientes propiedades según Selic(2003):

- ***Adecuado***: El modelo es suficiente para desarrollar una tarea. El modelo debe construirse con un propósito establecido, desde un punto de vista determinado y dirigido para usuarios específicos.
- ***Abstracto***: El modelo debe ser una versión reducida del sistema que representa, resaltando los aspectos más relevantes para su propósito ocultando los aspectos irrelevantes.
- ***Comprensible:*** El modelo debe ser expresado de tal forma que pueda entenderse fácilmente por sus usuarios.
- ***Preciso***: El modelo debe representar de manera exacta el sistema o realidad.
- ***Predictivo:*** El modelo se puede utilizar para responder preguntas sobre el sistema e inferir conclusiones correctas.
- ***Bajo costo:*** debe ser más fácil y económico de construir y estudiar que el propio sistema.

Así, la forma recursiva de definir modelos conforme a otros modelos con un mayor grado de abstracción culmina cuando se alcanza el nivel de meta-meta-modelo, debido estos se pueden considerar conformes a ellos mismos.

Para poder comprender la relación entre modelos y meta-modelos es común presentar una arquitectura de cuatro (4) niveles propuesta por la Object Management Group (OMG, 2014):

- ***M3 - Nivel de meta-meta-modelo***
  Este nivel caracteriza los meta-meta-modelos que describen los meta-modelos del nivel M2.
  Algunos ejemplos de lenguajes en este nivel son: Meta-Object Facility (MOF) usado por OMG (2007) y ECore usado por Eclipse Modeling Framework (EMF) (Steinberg et al., 2009), por enunciar solamente un par de ellos; que se usan para definir, construir y manejar elementos comunes a cualquier meta-modelo.

- ***M2 - Nivel del meta-modelo***
  Este nivel caracteriza los meta-modelos que describen los modelos del nivel M1, especificando las entidades de un lenguaje de modelado. Contempla un lenguaje de modelado general como UML o específicos (DSL). Como lo plantea OMG (2019), un meta-modelo especifica la sintaxis abstracta de un lenguaje de modelado, por lo cual es un tipo especial de modelo. Se puede entender como la especificación del conjunto de todos los modelos posibles expresados en ese lenguaje de modelado. Asimismo, se definen meta-modelos para otros propósitos como objetos de negocio, flujos de trabajo y modelos de componentes(Juan Bernardo Quintero & Anaya, 2007).

- ***M1 - Nivel de modelo***
  Este nivel incluye los modelos, que representan los datos, objetos y diseños de aplicaciones del nivel M0. Esto es, la descripción o especificación de un sistema realizado con un lenguaje determinado, a partir de su representación abstracta, mediante modelos; los cuales incorporan, por ejemplo, diseños de aplicaciones, estos son instancias de los metamodelos y constituyen el nivel M1.

En términos de programación orientada a objetos, este nivel corresponde a las definiciones de las clases de objetos de M0. En UML, cualquier diseño concreto de una aplicación, un modelo de proceso comercial, o un modelo de simulación, son ejemplo de este nivel.

- ***M0 - Nivel de objeto/instancia***
  Este nivel caracteriza los objetos o instancias del mundo real representados por el modelo, lo que sería la implementación de los diseños. Objetos de las clases en ejecución, en términos de programación orientada a objetos.

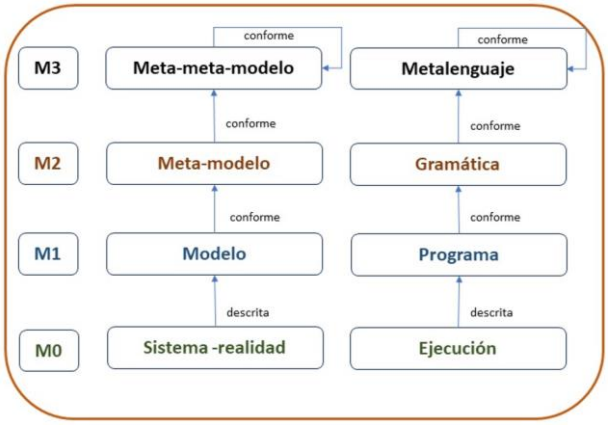


Figura 15. Representación arquitectura meta-modelado

Para mayor facilidad en la comprensión de la relación entre modelo, meta-modelo y meta-meta-modelo, a la luz de los cuatro (4) niveles de la arquitectura, en la figura 15 se presenta una analogía con lenguajes y gramáticas de programación. Por ejemplo, la ejecución de un programa escrito en un lenguaje (sistema del mundo real) (M0), es determinada por un programa (M1) conforme a una gramática del lenguaje (M2), que a su vez es definida con un meta-lenguaje (M3).

2. Meta-modelo

Un meta-modelo es un modelo que sirve de base para la construcción de otro modelo. Aunque ambos son modelos, uno se expresa en términos del otro, es decir que, un modelo es una instancia conforme al otro modelo (Gronback, 2009). En este sentido, un meta-modelo es a su vez un modelo que representa modelos (Jesús García Molina et al., 2014) y se construye con la finalidad de apoyar la comprensión del modelo principal respecto a las relaciones de las variables de estudio y así, poder tener una explicación y organización más detallada de su funcionamiento, estructura, datos y dinámica (Jesús García Molina et al., 2014). Un meta-modelo describe de manera abstracta, la posible estructura de los modelos, define las construcciones de un lenguaje de modelado y sus relaciones, así como las limitaciones y reglas de modelado (Thomas Stahl et al., 2013).

Otra definición de meta-modelo consiste en una herramienta metodológica y conceptual, que permite generar modelos alrededor de un entorno específico. Con estos modelos definidos se procede a realizar la aplicación de los mismos en la realidad, logrando una instancia o personalización del mismo al entorno que se aplique.

El concepto de meta-modelo se fundamenta en la ontología como especificación del

conocimiento en el cual se tiene un entendimiento de un dominio específico, de forma compartida, realizando una abstracción de los elementos de los modelos que se pueden instanciar en la realidad, para permitir la ideación de nuevos modelos que retomen las características genéricas y las personalicen al ámbito según los requerimientos específicos, permitiendo a su vez, la resolución de problemas con la construcción o derivación de modelos específicos(Angel et al., 2004).

Ahora bien, para la representación del meta-modelo se tiene una estrutuctura de cuatro (4) elementos: *entidades* (conceptos o clases), cada una con un conjunto de *atributos*, *asociaciones* y *restricciones* que permiten representar modelos conceptuales. Un meta-modelo se puede representar utilizando un subconjunto de los diagramas de clases de UML, con el entendido de que el objetivo en este caso no es hacer ningún tipo de implementación(Villalobos, 2021).

A continuación, se describe cada uno de estos elementos:

- *Entidad:* Son conceptos del dominio de conocimiento y se identifican con un nombre representativo y que es único dentro del meta-modelo. Se utiliza como sintaxis una caja, siguiendo lo establecido en un diagrama de clases de UML.

- *Atributo:* Es una característica del concepto, tiene un nombre que lo identifica y que refleja la característica que representa y puede tener o no, asociado un tipo de dato que indica los valores que este atributo puede tomar. Los atributos se sitúan dentro de la caja del concepto.

- *Asociación:* Son las relaciones existentes entre conceptos.

- *Restricción:* Pueden ser de dos tipos, el primero tipo se relaciona a las restricciones sobre los valores que pueden tomar los atributos de un concepto. Esta información no requiere incorporar algún elemento adicional dado que se incluye en la descripción del atributo. Por otro lado, el segundo tipo de restricciones está asociado a la estructura del modelo. Esta información se expresa en lenguajes formales o en lenguaje natural.

Un meta-modelo se construye partiendo de su raíz, ubicando un concepto que representa el dominio completo o el dominio a un alto nivel, que no es otra cosa que la realidad que se pretende modelar.

Posteriormente se desciende, a través de un proceso recursivo de descomposición de cada subdominio, en subestructuras de conceptos. Estos subdominios estarán conectados desde la raíz a través de una o más relaciones y tendrán relaciones entre conceptos que pertenecen a distintos subdominios.

La figura 16 muestra estos elementos mediante un ejemplo típico de la construcción de un meta-modelo, que lo constituye la definición de una máquina de estado simple (MES) que no tenga estados anidados ni otros aspectos avanzados.

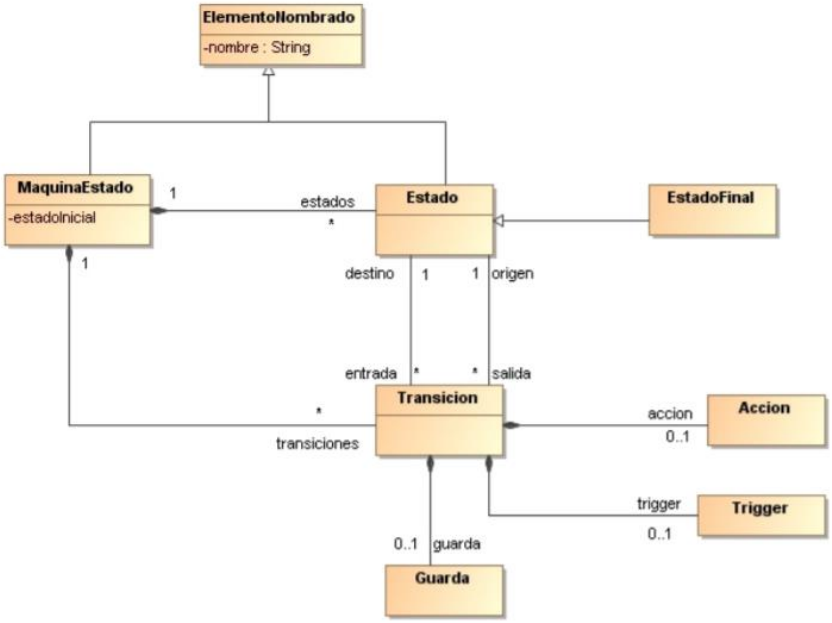


Figura 16. Meta-modelo que representa una máquina de estado simple -MES (Jesús García Molina et al., 2014)

De acuerdo con la figura, los elementos del lenguaje son:

- Entidades: ElementoNombrado, MaquinaEstado, Estado, Transicion, Guarda, Trigger, Accion y EstadoFinal.
- Atributos: nombre en ElementoNombrado; estadoinicial en MaquinaEstado.
- Asociaciones: MaquinaEstado con Estado y Transicion; Estado con Guarda, Trigger y Accion.

Es importante destacar, que un meta-modelo es el modelo conceptual que define la *sintaxis abstracta* de un dominio, mientras que el lenguaje utilizado expresa los modelos conformes con el meta-modelo, es decir su *sintaxis concreta*.

Para lo anterior, el lenguaje debe asociar una notación con los conceptos, atributos y relaciones del meta-modelo, tal como se muestra en la figura 17:

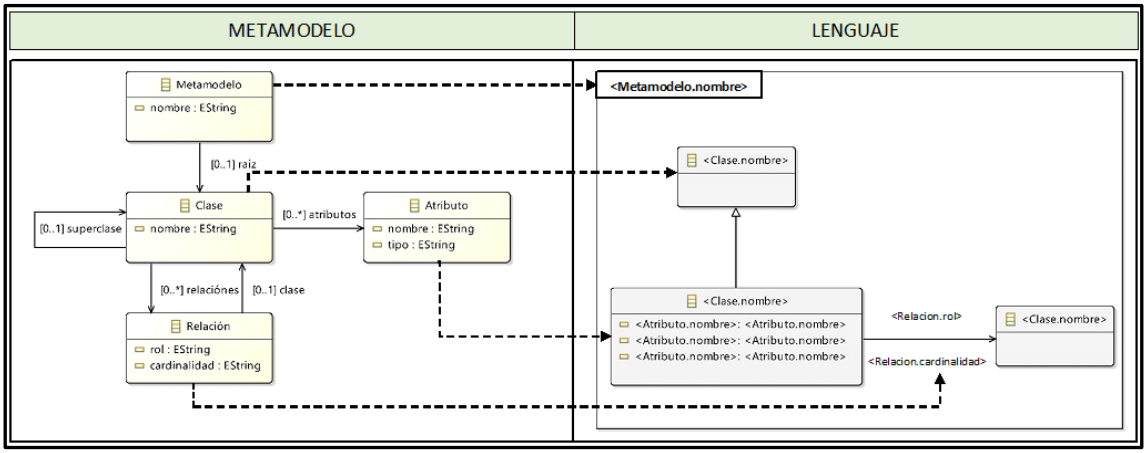


Figura 17. Ejemplo sintaxis abstracta vs sintaxis concreta

En la figura se puede verificar que cada concepto tiene una notación asociada y que para el concepto *atributo* tiene una notación, dentro de la notación del concepto *clase*. Asimismo, se utiliza el concepto *atributo* para indicar el atributo de un concepto, cada uno con un nombre y un tipo; y cada clase origina un conjunto de relaciones (tienen rol y cardinalidad) que se asocian a otra clase.

Si bien, la raíz es la clase *metamodelo*, que tiene un único atributo que es su nombre, en un diagrama no se indica cuál es la clase que corresponde a la raíz.

Para finalizar, los modelos se expresan utilizando lenguajes de modelado con un mayor nivel de abstracción que los lenguajes de programación habituales, con el propósito de tener un mecanismo común que permita la interoperabilidad de los sistemas mediante la transformación de un modelo a otro. Esto quiere decir que los lenguajes manejan conceptos más cercanos al dominio de la aplicación, dando la posibilidad de esta manera a que personas sin formación informática puedan desarrollar aplicaciones utilizando las herramientas adecuadas.

En Ingeniería basada en Modelos (en inglés Model-Driven Engineering) o MDE, los modelos son generalmente representados a través de enguajes de dominio específico (Domain specific languages) o DSL: y su meta-modelo especifica los conceptos del lenguaje, las relaciones entre ellos y las reglas estructurales que restringen los posibles elementos del modelo, así como aquellas combinaciones entre elementos que respetan las reglas semánticas del dominio (esta información, relacionada con MDE, se presenta en el *capítulo 8* de este documento).

CAPÍTULO 8. INGENIERÍA BASADA EN MODELOS

El modelado tiene un papel importante en el desarrollo de sistemas de software porque proporciona medios para abstraer conceptos en un dominio específico(Herrmannsdoerfer et al., 2009) y en ese sentido, la ingeniería basada en modelos (MDE) que surge como la respuesta a la industrialización del desarrollo de software, proporciona una mejor productividad y calidad (Vicente García-Díaz et al., 2009) y se convierte en un aliado al Blockchain que, con la promesa de transformar radicalmente la forma en que se intercambia valor (Nakamoto, 2008), se suma a las capacidades tecnológicas de las organizaciones y las habilita para lograr niveles cada vez más acelerados de productividad e innovación (Antonio et al., 2018).

1. Definición y terminología

La Ingeniería basada en Modelos (en inglés Model-Driven Engineering) o MDE, es un enfoque para el diseño y desarrollo de modelos para desarrollar software(B Selic, 2003), permitiendo una abstracción y formalismos de diferentes actividades del ciclo de desarrollo de software(Hailpern & Tarr, 2006), más adecuados y parecidos a los modelos mentales y con ello al dominio del problema, que los lenguajes de programación convencionales(Schmidt, 2006). Esta abstracción ayuda a que los usuarios finales puedan interactuar con el sistema utilizando los conceptos más comunes de estructura, entrada y salida, de forma textual o gráfica(Bapty & Sztipanovits, 1997), eliminando la información no relevante para su propósito. Por su parte, la información relevante del sistema es capturada por el MDE en un modelo específico del dominio de manera que se muestre de forma familiar a los usuarios del sistema (Thomas & Barry, 2003).

La clave de MDE es transformar modelos (que están en un nivel de abstracción alto), en modelos específicos de la plataforma por medio de herramientas que puedan transformar los modelos ejecutables en código fuente (Al-Batran et al., 2012), mediante un proceso automático (B Selic, 2003), evitando una fase que consume tiempo, recursos y que puede introducir errores (Balasubramanian et al., 2006).

La MDE es el último paso en el nivel de abstracción, en la que los modelos, que son la simplificación de un sistema con un objetivo previsto (Bézivin, 2005), se consideran los artefactos clave, ya que son los directores de todo el proceso de desarrollo del software(Núñez-Valdez et al., 2016).

Ingeniería dirigida por modelos (MDE) se caracteriza por (Trejo & Robles, 2010):

a. Poner al frente el nivel de abstracción ocultando los detalles específicos en la plataforma; ya que se elimina la etapa de codificación de las fases del desarrollo de sistemas.
b. Aprovechar el uso de los modelos en todas las fases de desarrollo de software para mejorar la comprensión.

c. Desarrollar framework y lenguajes específicos para lograr la comprensión del dominio.
d. Obtener provecho de las transformaciones para automatizar trabajo repetitivo y mejorar la calidad del software.

Así, la MDE resuelve los retos de los sistemas actuales altamente cambiantes y conectados, tanto en reglas de negocio como en tecnología, proponiendo un marco de trabajo para una arquitectura que asegura (Armas, 2012):

- *Portabilidad*, aumentando el re-uso de las aplicaciones y reduciendo el costo y complejidad del desarrollo y administración de las aplicaciones.
- *Interoperabilidad entre plataformas*, usando métodos rigurosos para garantizar que los estándares basados en implementaciones de tecnologías múltiples, tengan idénticas reglas de negocio. Por lo anterior, el desarrollo de modelos permite la generación de otros modelos que luego al ser juntados, proveerán la solución a todo un sistema e independiza el desarrollo de las tecnologías empleadas.
- *Independencia de plataforma*, reduciendo el tiempo, costo y complejidad asociada con aplicaciones desplegadas en diferentes tecnologías.
- *Especificidad del dominio*, a través de modelos específicos del dominio, que permiten implementaciones rápidas de aplicaciones nuevas, en una industria específica sobre diversas plataformas.
- *Productividad*, permitiendo a los desarrolladores, diseñadores y administradores de sistemas usar lenguajes y conceptos con los que se sienten cómodos, facilitando la comunicación e integración transparente entre los equipos de trabajo.

Como sucede con cada tecnología, hay una serie de términos que se deben entender en conjunto, para comprender el funcionamiento de esta. La figura, ilustra la arquitectura de los conceptos de MDE y en seguida, se describen.

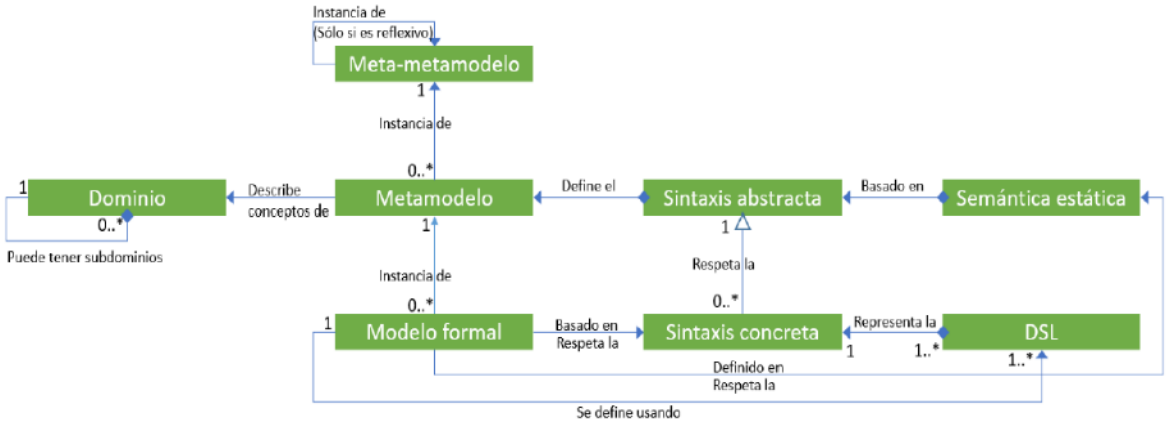


*Figura: 18. Arquitectura de los conceptos de M*DE (Gonzalez, 2017)

- **Sistema:** conjunto de partes y sus relaciones que pueden ser organizados para lograr un propósito. Así, un sistema puede ser un sistema hardware o software, una

compañía, procesos de negocio o la combinación de diferentes sistemas. Un sistema está formado por la plataforma y su aplicación (T. Stahl et al., 2013)

- **Dominio:** siempre es el punto inicial de MDE y delimita el campo de conocimiento. Así, el dominio es el área de conocimiento sobre la que se trabaja y sobre la que se quiere resolver el problema. Estos se dividen en dominios tecnológicos y profesionales, donde los primeros hacen referencia a la tecnología de desarrollo de software y los segundos a los conceptos que manejará la aplicación. Los dominios a su vez pueden estar compuestos de varios subdominios(Diaz, 2011).

- **Meta-modelo**: es una especificación del lenguaje de modelado que define las características del modelo y permite verificar el modelo expresado en ese lenguaje determinado de manera formal, es decir, es un modelo del lenguaje de modelado (Bran Selic, 2003a).

- **Meta-modelo reflexivo**: es cuando el meta-modelo de un lenguaje de modelado usa el mismo lenguaje de modelado, es decir, se define el meta-modelo utilizando el mismo lenguaje en el que el meta-modelo está descrito. Dentro de estos, el meta-modelo mínimo reflexivo es aquel que usa el mínimo número de elementos del lenguaje de modelado para los propósitos de ese meta-modelo, luego, si se eliminase cualquier elemento sería imposible modelar o expresar cualquier estado esencial (Gonzalez, 2017).

- **Meta-meta-modelo**: Es una especificación del meta-modelo que define las características del meta-modelo y permite verificar el meta-modelo expresado en ese lenguaje determinado, es decir, en un modelo del meta-modelo. El tener un meta-meta-modelo, permite que exista un meta-modelo para cada dominio del conocimiento a tratar mientras se tiene un meta-meta-modelo común a todos estos para así poder realizar operaciones sobre ellos, como pueden ser transformaciones automáticas, validaciones y búsquedas(Gonzalez, 2017)

- **Lenguaje de dominio específico (Domain specific languages -DSL)**: un DSL está constituido por la estructura, los términos, notaciones, sintaxis, semántica y reglas de integridad que son usadas para expresar un modelo. Algunos ejemplos de lenguajes de modelado son UML, SQL Schema, Business Process Management and Notation (BPMN), E/R, Ontology Web Language (OWL) y XML Schema (T. Stahl et al., 2013).

- **Punto de vista**: un punto de vista de un sistema es una técnica de abstracción en la que se seleccionan una serie de conceptos y reglas estructurales de ese sistema con el fin de centrarse en las preocupaciones importantes de ese sistema. Es decir, se crea una abstracción para suprimir detalles irrelevantes y así obtener un modelo simplificado de una parte del sistema (Kent, 2002). Cada punto de vista puede tener uno o más modelos, también conocidos como vistas.

- **Vista**: una vista es la representación del sistema, desde una perspectiva elegida de un punto de vista. Por ejemplo, si tenemos un sistema para mostrar los datos de los usuarios de un videojuego online, una vista podría ser, la forma en que está la información estructurada, otra vista la que muestra la información que puede ver

cada rol existente, otra vista podría ser, la que contiene los protocolos utilizados para transmitir la información y otra vista, para saber cómo se obtiene la información de los usuarios a partir de la información del sistema (Gonzalez, 2017).

- **Capas de arquitectura del meta-modelo**(Fuentes & Vallecillo, 2004): Arquitectura basada en cuatro niveles de abstracción que van a permitir distinguir entre los distintos niveles conceptuales que intervienen en el modelado de un sistema: M0– Datos; M1 – Modelo; M2 – Meta-modelo y M3 – Meta-metamodelo, presentados en el capítulo anterior.

2. Lenguajes de dominio específico (Domain specific languages -DSL)

Una de las ideas compartidas por los paradigmas englobados dentro del Desarrollo de Software Dirigido por Modelos (DSDM) es la conveniencia de que los programadores empleen lenguajes de más alto nivel de abstracción que los lenguajes de programación tradicionales, es decir, lenguajes que manejen conceptos más cercanos al dominio de la aplicación. Esto se hace a través de lenguajes textuales o visuales, necesarios para expresar los modelos a partir de los cuales se generan los artefactos de destino.

Estos idiomas son llamados lenguajes de modelado de dominio específico (DSL), ya que están diseñados para resolver problemas en un dominio y alcance específico(Bézivin, 2005a), por ello, estos lenguajes proporcionan mayor nivel de abstracción. A diferencia de lo que ocurre con los lenguajes de propósito general, gracias al uso de DSL, se consigue que los conceptos de un lenguaje se mapeen directamente a conceptos del dominio que se modela, sin posibilidad de interpretaciones erróneas.

En muchas situaciones del desarrollo de software se evidencia la repetitividad de los problemas, en esos casos se podrían solucionar utilizando un Lenguaje de Propósito General (General Purpose Language -GLP) como Java o C#, o se puede recurrir al uso de un DSL. Hay herramientas disponibles para definir DSL basados en meta-modelos, que facilitan la definición de una notación (sintaxis concreta) para el meta-modelo (sintaxis abstracta), que puede ser textual o gráfico (o una combinación de ambos).

Según (V García-Díaz, 2011), se pueden distinguir tres (3) clasificaciones para los DSL: 1. Desde el punto de vista de la construcción del lenguaje; 2. desde el formato de lenguaje y; 3. desde el dominio del problema. A continuación se especifica cada una de estas clasificaciones:

- *Desde un punto de vista de la construcción del lenguaje*:

Internos: Utilizan un determinado lenguaje anfitrión para darle la apariencia de otro lenguaje concreto. Un ejemplo son lo que actualmente se conocen como Fluent Interfaces.
Externos: Tiene su propia sintaxis y es necesario un parser para poder procesarlos. Un ejemplo de DSL externo es SQL (Structured Query Language)

- *Desde el punto de vista del formato del lenguaje:*

Textuales: La mayoría de los lenguajes informáticos son textuales y están formados por

un conjunto ordenado de sentencias. Un ejemplo de DSL textual es SQL utilizado para realizar consultas a una base de datos. Una forma de crear DSL textuales es mediante la creación de una determinada gramática (por ejemplo utilizando EBNF) y posteriormente crear o utilizar un parser para dicha gramática, para interpretar el DSL o generar código, en etapas posteriores.

Gráficos: En los últimos años están ganando gran aceptación los lenguajes gráficos, como UML. La creación de un lenguaje gráfico es similar a la de un lenguaje textual, la única diferencia es que en lugar de usar texto para representar los conceptos, se utilizan conectores y figuras simples.

- *Desde el punto de vista del dominio del problema:*

Horizontales: Los DSL horizontales son aquellos en los que el cliente que utilizará el lenguaje no pertenece a ningún dominio específico. Un ejemplo son los editores visuales de entornos de desarrollo que permiten generar interfaces de usuario automáticamente (por ejemplo Windows Forms de visual Studio).

Verticales: A diferencia de los DSL horizontales, el cliente que utilizará el lenguaje pertenece al mismo dominio que el lenguaje en sí. Como en el ejemplo anterior para un lenguaje de definición de encuestas, los usuarios finales serían los expertos en estadística encargados de definir dichas encuestas.

BLOQUE III
Desarrollo del Meta-modelo y del prototipo

ÍNDICE DEL BLOQUE

CAPÍTULO 9. META-MODELO Y PROTOTIPO

Esta tesis doctoral pretende dar respuesta a la pregunta planteada en el *Capítulo 1.1*, para intentar verificar la hipótesis planteada en el *Capítulo 1.2* y verificar los objetivos presentados en el *Capítulo 1.3.*

Para comprobar la hipótesis se deben cumplir los objetivos general y específicos, planteados en esta tesis.

Por un lado, los objetivos específicos permiten abordar la hipótesis de forma más sencilla y acotada; por el otro, el objetivo general es el que en últimas, permitirá verificar el cumplimiento de la hipotesis de la tesis doctoral.

La solución general que da origen al meta-modelo propuesto, se presenta a continuación, y el prototipo se presenta en el siguiente numeral de este capítulo.

1. Meta-modelo

Como se ha mencionado, en esta investigación se realiza la revisión, elección y uso de herramientas que se soportan en tecnología blockchain pública, es decir que permite acceso sin restricción y prevalece el parámetro de transparencia (Bashir, 2017a), lo que permite que la participación sea abierta, sin que se pierdan los atributos de seguridad y transparencia; esto es justo lo que se requiere para el sector público.

Por consiguiente, se usa blockchain público y smart contracts aplicadas a la garantía de transparencia y de anticorrupción en el sector público, a la vez que se aplica en un caso concreto en el que el prototipo se destaca en temas de seguimiento transparente de la planificación y ejecución de presupuestos, ampliando el alcance a una solución general que sirva para cualquier aplicación en el sector público mediante un meta-modelo.

Aunado a lo anterior y considerando la necesidad de reducir los niveles de corrupción en la administración pública, se propone el diseño de un meta-modelo que permite la generación de un modelo activo e inteligente para hacer seguimiento a la contratación y a la ejecución de presupuestos en el sector público, dado que la ciudadanía exige una administración más transparente, rápida y eficiente.

Este meta-modelo de integración blockchain y smartcontracts para gestionar contratos en la administración pública se diseñó aplicando Ingeniería dirigida por modelos (MDE), lo que conlleva una serie de pasos, el inicial consiste en definir el dominio del problema que se quiere resolver, acotando el campo de acción.

En esta ocasión, el dominio es la creación y asociación de contratos a presupuestos de la administración pública, de manera que haya transparencia en la ejecución contractual y uso de los recursos definiendo y haciendo uso de reglas o consensos que permiten automatizar los avales de desembolso de presupuestos, previo cumplimiento de condiciones de ejecución.

Una vez que se tiene acotado el dominio, se definió el meta-modelo que permitirá crear los modelos de las soluciones específicas. El meta-modelo es reutilizable, interoperable y portable, esto dedibo a que el meta-modelo es una abstracción de alto nivel.

En la figura 19 se muestra el meta-modelo y está conformado por los siguientes elementos: **contrato inteligente** *(smart contract),* **listas de bloques** *(block list),* **web service, P2P** *y* **presupuesto público** *(public budget).*

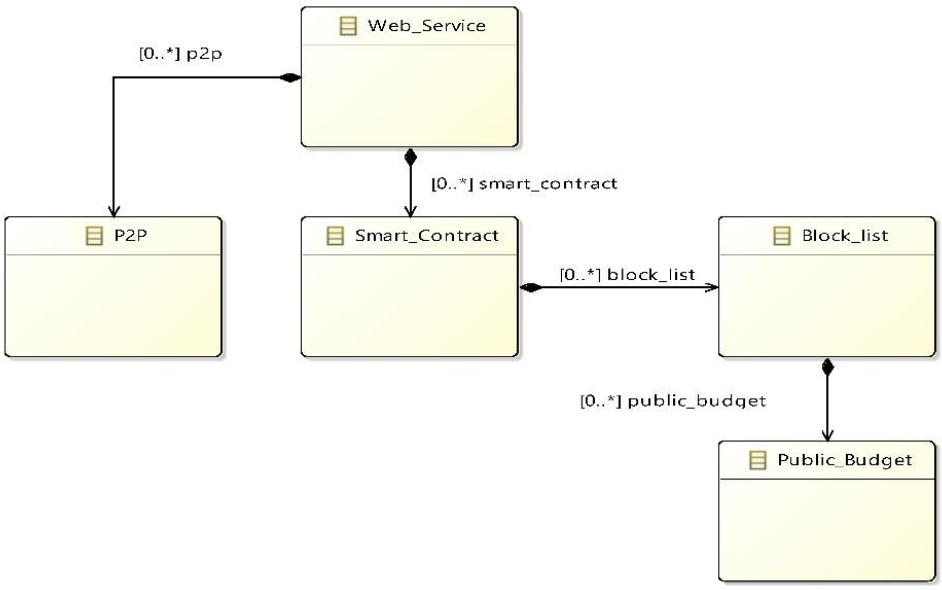


Figura: 19. Meta-modelo propuesto

Por un lado, la *lista de bloques (**block list**),* proporciona los datos relacionados con rubros y contratos, las cuales guardan relación directa con el *presupuesto público **(public budget**),* que es la cantidad disponible para cada entidad pública según los rubros asignados, que a través del *contrato inteligente (**smart contract**)* consensúa algunas reglas, para la ejecución automática de una o más instrucciones entre los actores, respecto de ejecución de contratos y de presupuestos. Se activa en el modelo, una vez determina que se cumplen las reglas previamente definidas.

Por otra parte, es por medio de *Peer-to-peer **(P2P)*** que se crea una red de bloques, la cual almacena en la nube cada una de las transacciones que se registren con los presupuestos y las listas de bloques, descentralizando la información, lo que admite la creación de acuerdos de smartcontracts basada en blockchain por lo cual no permite que exista falsificación de las transacciones. Lo anterior, se conecta mediante un *servicio web **(web service)*** haciendo posible la comunicación y el intercambio de datos entre diferentes servidores o aplicaciones, sin importar las diferencias que existan entre los lenguajes de programación en el que fueron desarrolladas o la plataforma en la que se ejecutan.

En consecuencia, el diseño de un meta-modelo que responda a una abstracción claramente definida del dominio de conocimiento abordado, y que atienda a requerimientos sugeridos, contribuirá a mejorar la transparencia y el control de la contratación publica a la luz de la ejecución presupuestal, además de proporcionar un conjunto de propiedades que permite y facilita procesos de registro y consulta por parte de los usuarios de las acciones de ejecución de contratos generando un mayor nivel de confiabilidad ofrecida a los usuarios.
A continuación, se describe brevemente el Modelo de integración de blockchain y smartcontracts para la gestión de contratos en el sector público (figura 20), como una

solución que esta tesis doctoral propone a partir del meta-modelo general:

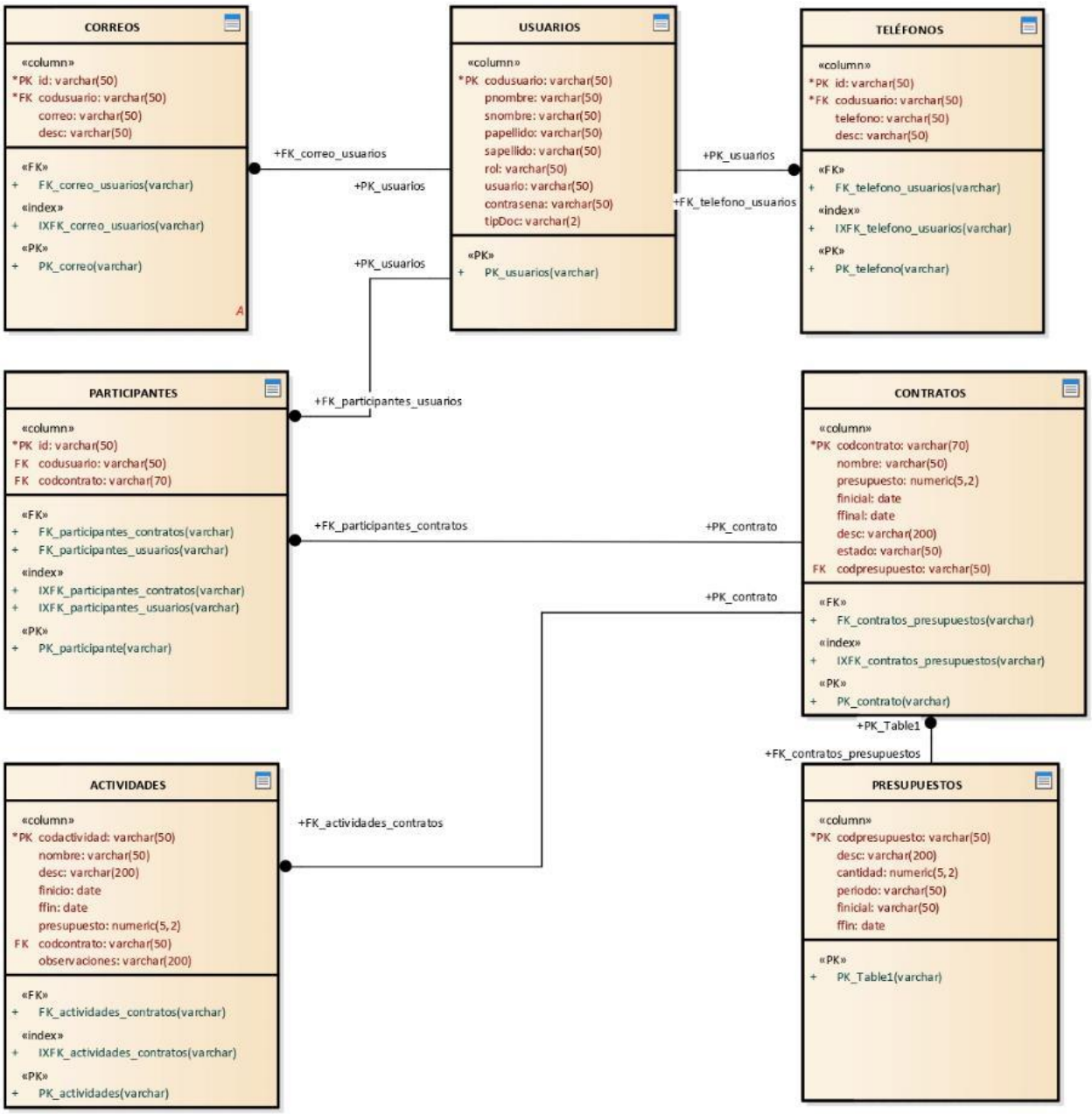


Figura 20. Modelo de integracion de blockchain y smartcontracts para la gestión de contratos en el sector público

- ***Participante:***

Es el componente principal y su definición es el objetivo del lenguaje. Sus propiedades son el tipo de rol que adquiere en la plataforma y su relación con los contratos que genere el empleador.

El participante es un usuario que se registra en la plataforma con información básica para efectos de firmar contratos, efectuar descargues de presupuestos, así como para notificar novedades.

Los actores participantes representan tanto a contratantes, contratistas como a auditores con capacidades de ejecución acciones de procesamiento de información (crear, editar, gestionar, aprobar contratos, etc).

- ***Contratos****:*

Un participante podrá tener asociados uno o más contratos, pero al menos uno. Por su parte, los contratos son un componente abstracto que otro componente debe implementar, es decir, participante puede tener varios contratos adjudicados con diferente nivel de ejecución y con distintas propiedades, pero todos son contratos. Para poder verificar la ejecución de cada contrato y con ello, la destinación de presupuesto, el contrato contiene actividades que constituye las fases, etapas o entregables que, en suma, propenden por el cumplimiento del objetivo para el cual fue requerido por la entidad pública.

- **Smart contract:**

Protocolo que al tener definidas algunas reglas o consensos, automatizan las ejecuciones de las acciones de los participantes respecto de las transacciones de ejecución de contratos y de presupuestos. El Smart contract se activa en el modelo, una vez determina que se cumplen las reglas previamente definidas.

- **Presupuestos:**

El presupuesto está constituido por dinero disponible por cada entidad pública y que se ejecuta mediante contratos. El presupuesto es definido mediante un plan anual de adquisiciones que contiene el valor de este, desagregado según su destinación para solventar necesidades de la entidad (agrupación por rubros). En el prototipo se cuenta con dos tipos de presupuestos, el primero como se indicó, corresponde al que se le asigna a la entidad pública y el segundo tipo corresponde al presupuesto asignado para la ejecución de un contrato de manera espefícica (subpresupuesto).

- **Actividades:**

Son las diferentes acciones que, en el marco del desarrollo y cumplimiento de los contratos suscritos, se realizan para garantizar la ejecución de los presupuestos.

- **Correos:**

Direcciones electrónicas para validar el tipo de usuario y los roles asignados. Util como una de las herramientas para verificación de autenticidad.

- **Usuarios:**

Participantes en el modelo, sin embargo, tendrán roles distintos según sea su forma

de participación.

- **Teléfonos:**

Dato para establecer alternativa de contacto o comunicación con los usuarios, adicionalmente es otra de las herramientas para verificación de autenticidad.

Aunado a lo anteriormente expuesto, se modela el *control o vigilancia fiscal*, realizado por entes de control del estado colombiano. Mediante este proceso se evalúan los resultados obtenidos por las diferentes entidades del Estado, al determinar si adquieren, manejan y/o usan los recursos públicos dentro del marco legal, sujetos a los principios de economía, eficiencia, eficacia, equidad y sostenibilidad ambiental.

A la vez que se busca establecer la responsabilidad fiscal de los servidores públicos y de los particulares que causen, por acción o por omisión y en forma dolosa o culposa, un daño al patrimonio del Estado.

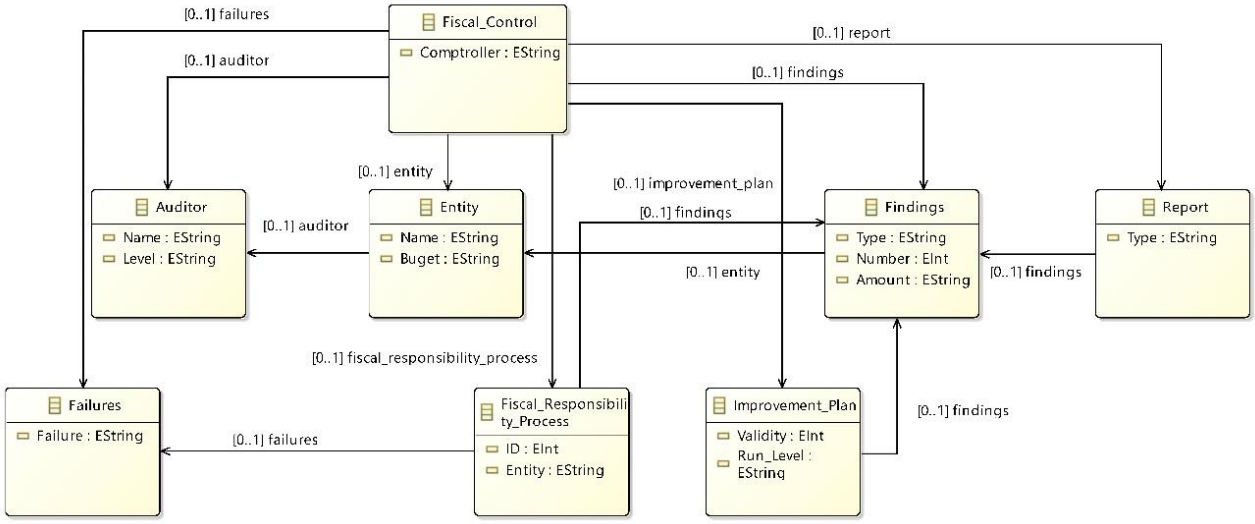


Figura 21. Modelo de control fiscal en el sector público (Triana Casallas, Jenny Alexandra, Rodriguez Molano, José Ignacio, Fuentes, 2020)*; 2020)*

El modelo presentado en la figura anterior, tiene ocho (8) metaclases, así:

- ***Control Fiscal**:*
  Ente de control que realiza la vigilancia o el control fiscal a las entidades u organizaciones que son de su competencia por el origen de los recursos ejecutados.

- ***Auditor:***
  Personal de nivel profesional o especializado que realiza la vigilancia o el control fiscal a través de auditorías.

- ***Entidad:***
  Organización (generalmente pública) que ejecuta los recursos públicos asignados.

- ***Hallazgos:***
  Resultados de las auditorías de control o vigilancia fiscal, se originan por detección de irregularidades en la ejecución de los recursos públicos asignados a las entidades, generalmente en temas de contratación. Suelen ser de tipo administrativo, fiscal, disciplinario, sancionatorio o penal.

- ***Informe:***
  Es el documento en el que se describe el proceso de auditoría de control o vigilancia de los recursos públicos, incluye: planificación, auditoría in situ y hallazgos. Este informe puede ser preliminar (con derecho a réplica) o definitivo.

- ***Plan de Mejoramiento****:*
  Como acción correctiva frente a los hallazgos comunicados en el informe definitivo, se establecen los planes de mejoramiento para cada vigencia auditada.

- ***Proceso Responsabilidad Fiscal:***
  Si los hallazgos se configuran como fiscales, se dará apertura a un proceso de responsabilidad fiscal, que constituye la determinación de la responsabilidad de los servidores públicos y de los particulares, cuando ejecutando el presupuesto de las entidades, causen daño al patrimonio del Estado.

- ***Fallo:***
  Última etapa, es una orden o sentencia. Los fallos pueden ser con responsabilidad fiscal, sin responsabilidad fiscal, o archivados por no mérito.

En este sentido, el modelo de integración blockchain y smartcontracts para gestionar contratos en la administración pública puede facilitar la labor del control o vigilancia fiscal, teniendo en cuenta que al accecer al estado actual de la gestión contractual de las entidades, en concordancia con los presupuestos anuales aprobados para cada entidad pública que por supuesto, se constituyen como sujetos de control fiscal.

2. Prototipo

Con base en el meta-modelo expuesto anteriormente, se inicia el análisis de posibilidades para llegar a la creación de un prototipo que permite crear, aprobar, consultar y auditar transacciones, de manera autónoma y transparente, entre contratante (entidad pública) y contratistas en procesos de contratación en el sector público.

Más adelante se notará como este prototipo fue probado y operado en la Contraloría Municipal de Villavicencio (CMV), una entidad pública de Control Fiscal que vigila la gestión fiscal de las entidades públicas y de particulares que ejecutan recursos originados por el Municipio de Villavicencio.

El prototipo propuesto está basado en blockchain, en el cual las transacciones originan un historial transparente a la vista de los actores de la cadena, ofreciendo la certeza de que la información no ha sido alterada. Lo anterior, en virtud de que la blockchain permite registrar y publicar datos sin necesidad de intermediarios que centralicen la información. A la vez, con la incorporación de smartcontracts a la blockchain, se tiene la posibilidad de autorizar la transferencia de recursos públicos con información validada en tiempo real.

Este prototipo facilita la auditoría interna y de los organismos de control, además de coadyuvar en la mejora de la transparencia y la confiabilidad en los procesos de contratación de las entidades.

Lo anterior debido a que la tecnología blockchain cuenta con indudables atributos y beneficios para generar confianza digital, así como mayor eficacia y eficiencia en las operaciones y la gestión de procesos en las organizaciones, entre las que se encuentran:

- Inmutabilidad de los registros
- Seguridad de la Información
- Trazabilidad
- Encadenamiento
- Descentralización de los datos
- Transparencia
- Desintermediación

Aunado a lo expuesto, al integrar smartcontract a la plataforma bajo tecnología blockchain, se logra que esta adquiera también las siguientes características:

- Autonomía
- Confiabilidad
- Autosuficiencia
- Seguridad
- Rapidez
- Exactitud
- Descentralización

En consecuencia, debido a la unión y potencialización de las características enunciadas y descritas en el bloque teórico de este documento; el prototipo propuesto que integra blockchain y smart contracts para la contratación pública adquiere atributos importantes para abordar las problemáticas asociadas a la falta de transparencia de este tipo de procesos. Entre estos atributos más importantes se pueden enunciar:

- Las transacciones son auditables
- Los presupuestos se puedan verificar a través de la cadena de desembolso vs el cumplimiento de actividades contractuales acordadas entre las partes.
- Los registros de transacciones no pueden ser alterados

Ahora bien, al definir el sistema o el prototipo, se estableció que las propiedades

mínimas que este debe tener, serán las que le otorga el hecho de integrar smartcontracts a blockchain (se presentaron en los párrafos anteriores y en el bloque teórico de la tesis) y de manera específica para el caso, las siguientes: por un lado, que permita el almacenamiento encriptado transacciones asociadas al proceso de formalización y ejecución de actividades contractuales al interior de la entidad pública, mitigando riesgos de manipulación, ajustes no consensuados y de pagos por actividades no ejecutadas, no cumplidas o con cumplimiento en el plazo acordado para su culminación.

Convirtiéndose este, en una herramienta de interés para el control de los recursos y del cumplimiento de las actividades pactadas, en tiempo real, sin una entidad intermediaria que podría modificar las condiciones pactadas, alterar el contenido de la transacción e incluso ocasionar demoras en la validación de la información y por tanto, de los desembolsos respectivos.

2.1. Requerimientos para el diseño del prototipo

El prototipo es el resultado de los lineamientos conceptuales y las fases metodológicas señaladas en la primera parte de este documento. Para su desarrollo se requirió articular conocimientos, tecnologías y tiempos.

A continuación, se enuncian las variables identificadas para apoyar el diseño y desarrollo del sistema:

a) Roles y funcionalidades:

Se identificaron los actores que intervienen dentro del proceso, derivando de allí, funcionalidades para el sistema y la aplicación de blockchain.

- *Contratante*

Persona encargada de elaborar los contratos, destinación de presupuesto y delimitación de actividades. Puede visualizar el avance de un contrato y de ser necesario bloquearlo ante un incumplimiento.

- *Contratista*

Usuario que puede visualizar el contrato, firmar el contrato, reportar actividades realizadas o inconvenientes que pueda tener.

- *Auditor*

Encargado de revisar por qué un contrato fue bloqueado y decide acerca de qué acciones se deben seguir.

Acciones	Actores/Roles
Registro de usuarios	• Contratante • Contratista • Auditor
Acceso de usuarios	• Contratante • Contratista • Auditor

Construcción de contratos	• Contratante
Gestión de presupuesto	• Contratante
Edición de contratos	• Contratante
Aceptación de contrato	• Contratante • Contratista
Avance de cumplimiento del contrato	• Contratante
Reporte de acciones realizadas del contrato	• Contratista
Bloqueo de contrato	• Contratante
Generación de contrato nuevo	• Contratante
Investigación de transacciones	• Auditor

Tabla 7: Roles o actores y funcionalidades para el prototipo

Los casos de uso, su priorización y especificación, se pueden consultar en el *Anexo 1: Análisis del sistema*.

b) Lista de campos de MetaData de los contratos y el presupuesto de la entidad para selección e incorporar al Smart contract:

Los campos que se presentan corresponden a los valores necesarios para establecer un control de los presupuestos ejecutados y cómo se gastaron.

Campos de metadatos para incorporación del smartcontract
<<Presupuesto>>
<<Contrato>>
<<Presupuesto para el contrato>>
<<Valor del contrato>>
<<Fecha de inicio del contrato>>
<<Fecha de fin del contrato>>
<<Actividad del contrato>>
<<Actividad del contrato>>
<<Valor de la actividad del contrato>>
<<Fecha de inicio de la actividad del contrato>>
<<Fecha de fin de la actividad del contrato >>
<<Firma contratante>>
<<Firma contratista>>
<<Reporte de actividad>>
<<Seguimiento de actividad>>
<<Aprobación de actividad>>

Tabla 8: Campos de metadatos para incorporación del smartcontract

Con los anteriores requerimientos, se diseñó una arquitectura basada en Blockchain que permitiera abordar los requerimientos de construcción, edición, aceptación avance y reporte de cumplimiento y bloqueo de contratos, gestión de presupuesto e investigación o auditoría de transacciones, con su respectivo vínculo entre presupuesto y contrato.

Para el prototipo se eligió PostgreSQL, que es un sistema de gestión de bases de datos relacional orientado a objetos y de código abierto, Psycopg2 que es un controlador PostgreSQL compatible con DB API 2.0 que se desarrolla activamente y las redes P2P.

En ese diseño, se resaltan unos componentes principales que se asocian y son la columna vertebral del prototipo: redes P2P, smartcontract y PostgreSQL, los cuales se representan y se explican a continuación:

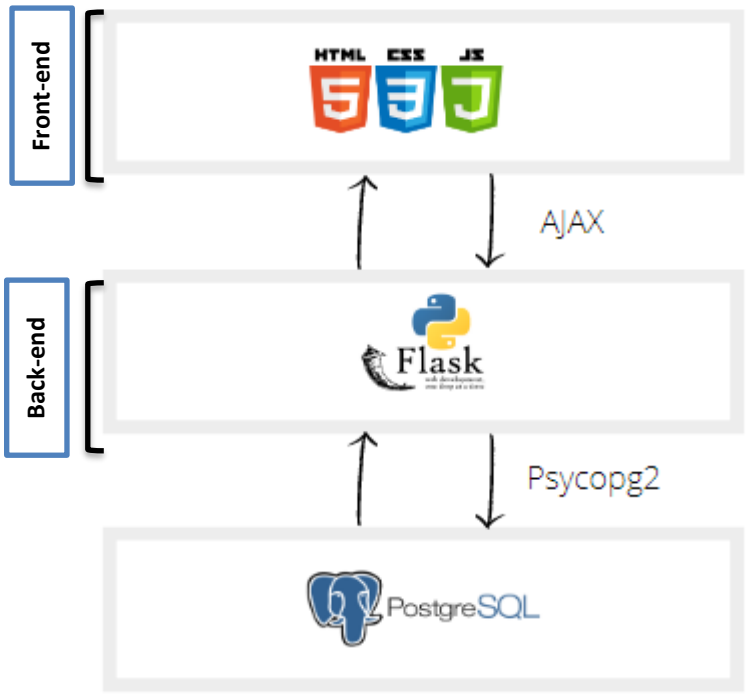


Figura: 22. Modelo general de la aquitectura del prototipo

***Peer-to-peer*:** fue elegido como opción viable para el desarrollo del prototipo, dado que por medio de P2P (peer-to-peer) se crea una red de bloques, que almacenará cada una de las transacciones que se registren en la plataforma, esta cadena inicia con un bloque denominado "Génesis" el cual contiene un código hash generado por una función hash única creada a partir del contenido del bloque y así cada bloque siguiente a este.

El objetivo de la conexión P2P es descentralizar la información, es decir, hacer a un lado la conexión tradicional a un único servidor. descentralizada que permite la creación de acuerdos de smartcontracts entre pares, basada en blockchain.

Cuenta con la ventaja que cualquier desarrollador puede crear y publicar aplicaciones distribuidas que realicen smartcontracts en esta plataforma y como se presentó en el bloque teórico de este documento, cuenta con una serie de características, muchas de ellas, pertinentes para este prototipo, permitiendo:

- Transacciones confiables y obtener trazabilidad.
- Desarrollar una interfaz de usuario integrada y funcional.
- Trabajar con redes públicas, privadas e híbridas

Se implementan ***smartcontracts (contratos inteligentes),*** para automatizar las actividades de un contrato, frente al presupuesto comprometido para su ejecución. El smartcontract lleva un control de los presupuestos gastados y cómo se gastaron.

PostgreSQL como solución al almacenamiento de la información de los actores y de las transacciones, toda vez que es un gestor de bases de datos relacional que provee fácil accesibilidad, es multiplataforma y está disponible para su uso en la mayoría de sistemas operativos, sin afectar su rendimiento. Por otro lado, se crea un método peer-to-peer direccionable por contenido para almacenar y compartir información en un sistema de archivos distribuidos.

A continuación, se presenta una tabla con los requerimientos funcionales y no funcionales del prototipo:

Requerimiento	Descripción	Tipo
Registro de usuarios	Por medio de un formulario registrar los datos de los usuarios que van a interactuar con la plataforma, se crea usuario y contraseña.	Funcional
Acceso de usuarios	A través de la página de login con el usuario y contraseña permitir el acceso de los usuarios a la plataforma.	Funcional
Delimitación de roles	Permite mostrar las funciones acordes a cada usuario, según sus permisos en la realización e interacción con los contratos.	Funcional
Construcción de contratos	Ayuda a la elaboración de los contratos, detallar cada una de las actividades, los participantes, el tiempo y el presupuesto que este requiere.	Funcional
Gestión de presupuesto	Se detalla el destino que tiene el presupuesto en los diferentes contratos y actividades de estos.	Funcional
Edición de contratos	Se puede corregir cualquier punto de un contrato antes de que este sea aceptado.	Funcional
Aceptación de contrato	Una vez todos los detalles del contrato estén en orden, la persona o personas encargadas podrán firmarlo para que empiece su ejecución.	Funcional
Avance de cumplimiento del contrato	Visualización del progreso de un contrato con respecto al tiempo que tiene este para ser culminado.	Funcional
Reporte de acciones realizadas del contrato	Detalle de cada una de las actividades realizadas de un contrato, estado, observaciones y presupuesto gastado.	Funcional
Alertas de incumplimiento	Si existen retrasos en tiempo o gastos fuera de los previstos, se generará una alerta a los encargados de un contrato.	Funcional
Bloqueo de contrato	Si es necesario se puede bloquear un contrato que este siendo incumplido.	Funcional
Resumen de contratos en ejecución	Se puede visualizar los contratos que se estén ejecutando, con un breve resumen de su objetivo principal y el presupuesto que requieren.	Funcional
Generación de contrato nuevo	Usando un contrato ya firmado que deba modificarse, se crea un nuevo contrato con los cambios necesarios.	Funcional
Historial de transacción	Cada una de las actividades realizadas dentro de la plataforma serán registradas en el sistema blockchain.	Funcional
Investigación de transacciones	Se puede revisar el historial de las transacciones para hacer un seguimiento de un contrato, delimitación de presupuesto o de un usuario.	Funcional
Acceso a la plataforma	El acceso a la plataforma de los contratos inteligentes es por medio de una interfaz web disponible para diferentes dispositivos.	No funcional
Interfaz gráfica de usuarios	La GUI por medio de menú de navegación, ventanas, formularios y botones ayudará a los usuarios a hacer de manera más fácil los procesos.	No funcional
Base de datos	Se hará uso de una base de datos relacional mediante un manejador convencional.	No funcional
Seguridad para los usuarios y contratos	La seguridad de los usuarios y los contratos deberá ser manejada desde la base de datos, haciendo uso de algoritmos de encriptamiento y determinación de roles.	No funcional

Tabla 9: Requerimientos del sistema

2.2. Características del prototipo

El prototipo tiene las siguientes características:

- *Automatización y desintermediación en los procesos:* gracias a la integración de smartcontracts con la tecnología blockchain, debido al mecanismo del consenso distribuido proporcionado por esta última.

- *Recopilación y procesamiento de datos en cualquier momento:* El prototipo no restringe la captura de información y las auditorías pueden ejecutarse en cualquier momento.

- *Interfaz de usuario*: el prototipo puede comunicarse con el usuario a través de un menú de navegación, ventanas, formularios y botones.

- *Comunicación entre los actores sin intermediarios:* el sistema no requiere intervención de una autoridad central para operar. La comunicación se realizada bajo los principios de las redes peer-to-peer, debido a que estas permiten el intercambio directo de información, en cualquier formato, entre los ordenadores interconectados, sin la necesidad de servidores fijos, es más, actúan simultáneamente como clientes y servidores respecto a los demás nodos de la red.

- *Seguridad para los usuarios y contratos:* La seguridad de los usuarios y los contratos deberá ser manejada desde la base de datos, haciendo uso de algoritmos de encriptamiento y determinación de roles.

- *Disposición total de datos:* el sistema genera una red de comunicación que permite el acceso y consulta en tiempo real.

2.3. Funciones del prototipo

Los diagramas que siguen, presentan las funciones generales que ofrece el prototipo:

- Elaboración de contratos

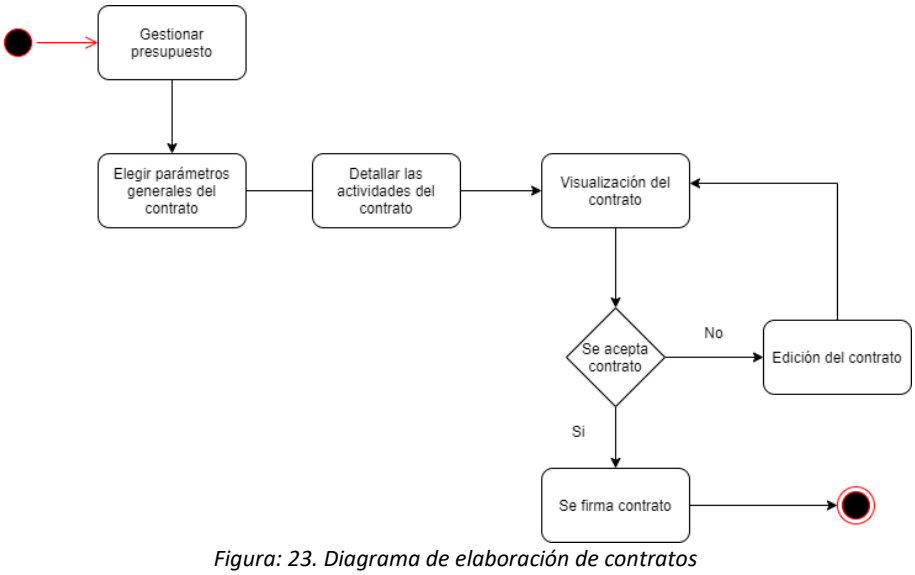


Figura: 23. Diagrama de elaboración de contratos

- Ejecución de contratos

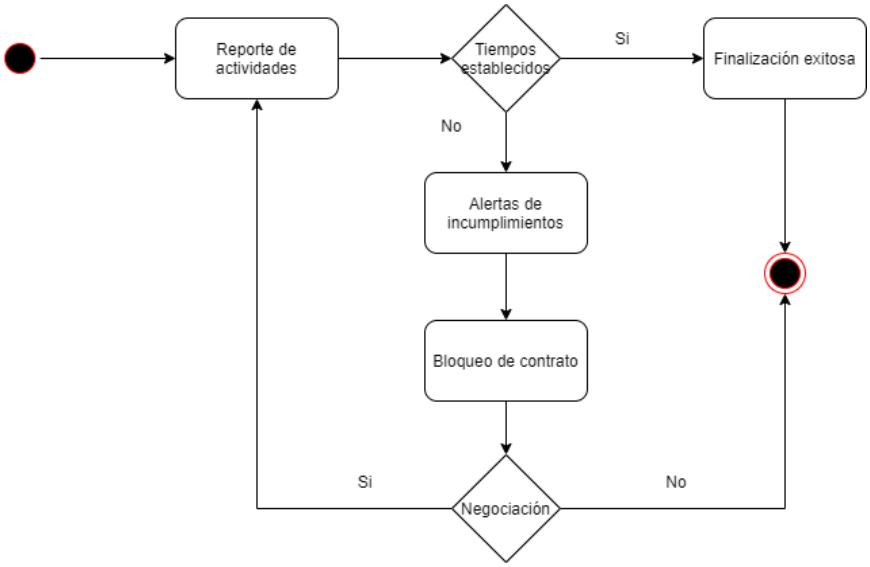


Figura: 24. Diagrama de ejecución de contratos

- Gestión de contrato nuevo

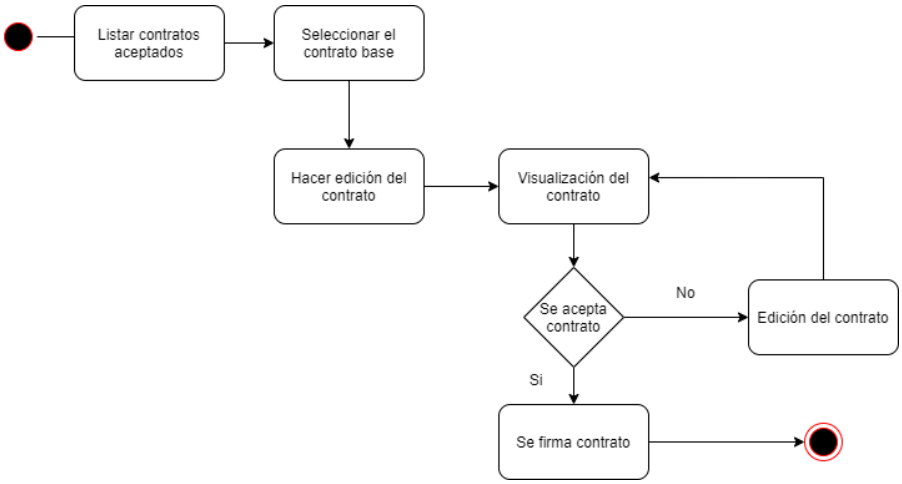


Figura: 25. Diagrama de gestión de contrato nuevo

- Auditoría

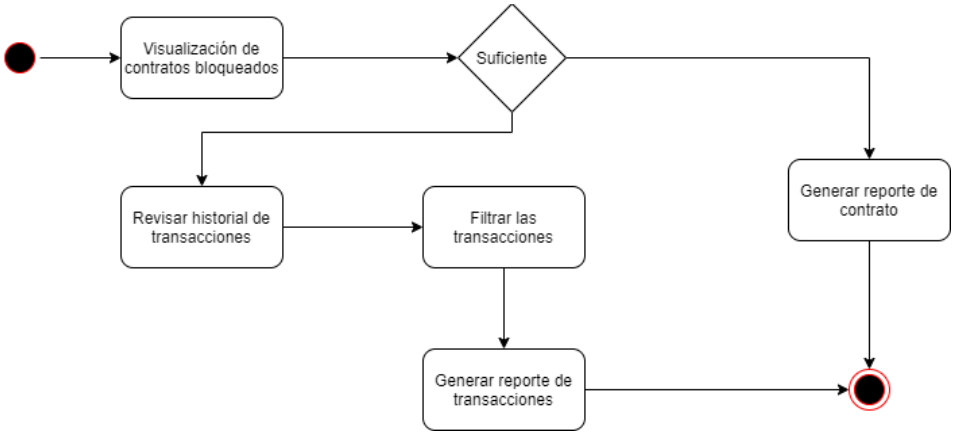


Figura: 26. Diagrama de auditoría

CAPÍTULO 10. VALIDACIÓN, PRUEBAS Y RESULTADOS

Con el propósito de validar el cumplimiento del objetivo del modelo propuesto, se efectuaron pruebas de operación con el prototipo, implementado en la Contraloría Municipal de Villavicencio (CMV), una entidad pública de Control Fiscal territorial creada mediante Acuerdo No. 038 de junio 6 de 1986, que tiene como fin ejercer la vigilancia de la gestión fiscal del Municipio de Villavicencio (Meta) y de particulares que manejen o administren bienes o recursos del municipio.

Retomando, la aplicación de pruebas en el prototipo se realiza con base en la contratación programada y realizada por la Contraloría Municipal de Villavicencio (CMV) -Colombia en 2021, evaluando el desempeño del prototipo con sus elementos constitutivos.

En primer lugar, se realizan pruebas de usabilidad y, en segundo lugar, se valida información frente a hechos de corrupción o problemas de transparencia en la gestión contractual de la CMV antes y después de usar el prototipo.

Es importante aclarar que el punto de partida del prototipo lo constituye el Plan Anual de Adquisiciones (PAA) que es un instrumento de planeación contractual de las Entidades Estatales Colombianas para: (i) facilitar a las Entidades Estatales identificar, registrar, programar y divulgar sus necesidades de bienes, obras y servicios; y (ii) diseñar estrategias de contratación basadas en agregación de la demanda que permitan incrementar la eficiencia del proceso de contratación. Su publicación en la web de la entidad pública es una obligación legal (artículo 3 del Decreto 1510 de 2013, compilado en el Decreto 1082 de 2015).

El PAAC debe elaborarse cada año con alcance desde el 01 de enero hasta el 31 de diciembre de cada año.

A continuación, se presenta la pantalla antes de incorporarse la información de los presupuestos disponibles por la entidad para realizar la contratación, lo anterior, contemplado en el documento denominado Plan Anual de Adquisiciones (PAA) que es un instrumento de planeación contractual de la Entidad Estatal para: (i) facilitar a las Entidades Estatales identificar, registrar, programar y divulgar sus necesidades de bienes, obras y servicios; y (ii) diseñar estrategias de contratación basadas en agregación de la demanda que permitan incrementar la eficiencia del proceso de contratación. Su publicación en la web de la entidad pública es una obligación legal (Decreto Único Reglamentario 1082, 2015).

1. Pruebas de operación

Acceso y roles

La plataforma funciona a través de un servidor web de manera que cualquier usuario registrado pueda acceder fácilmente sin necesidad de instalaciones o requerimientos previos que puedan poner en riesgo la funcionalidad del sistema, simplemente se necesita un

navegador y acceso a internet.

Ahora bien, para acceder por primera vez, se registran los datos de los usuarios que van a interactuar con la plataforma, se crea usuario y contraseña, posteriormente se accede en la ventana de login, con el usuario y contraseña, si esta es incorrecta el acceso será denegado.

A continuación en la figura 27, se presenta captura de pantalla de la interfaz inicial:

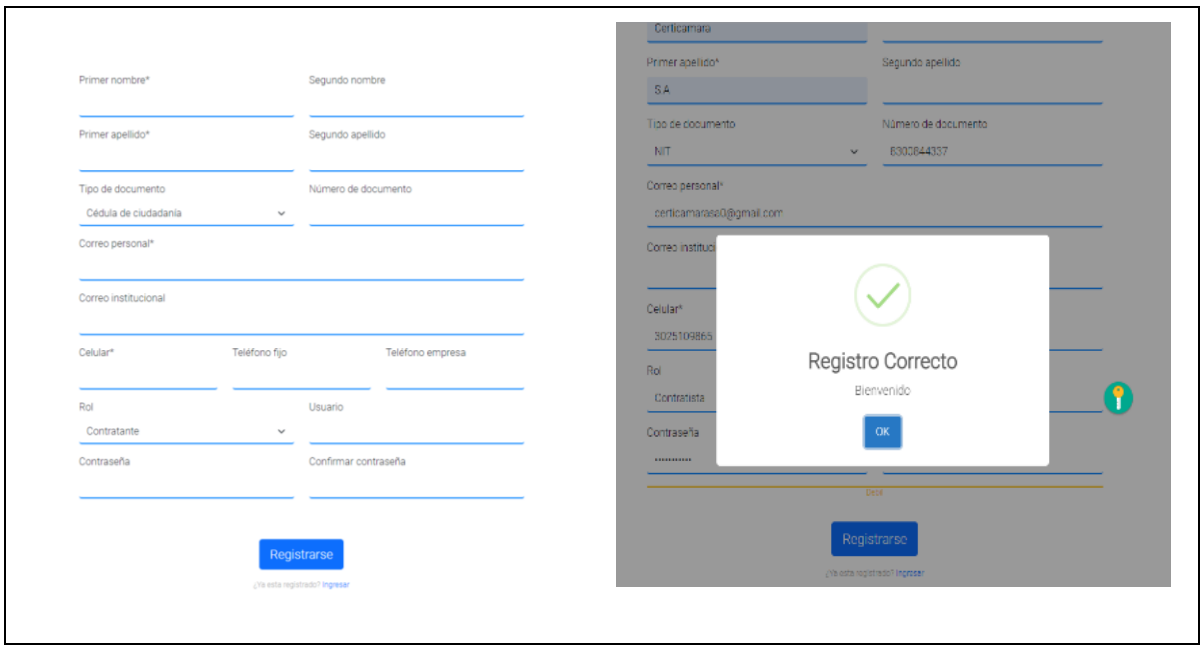


Figura: 27. Ventana de registro

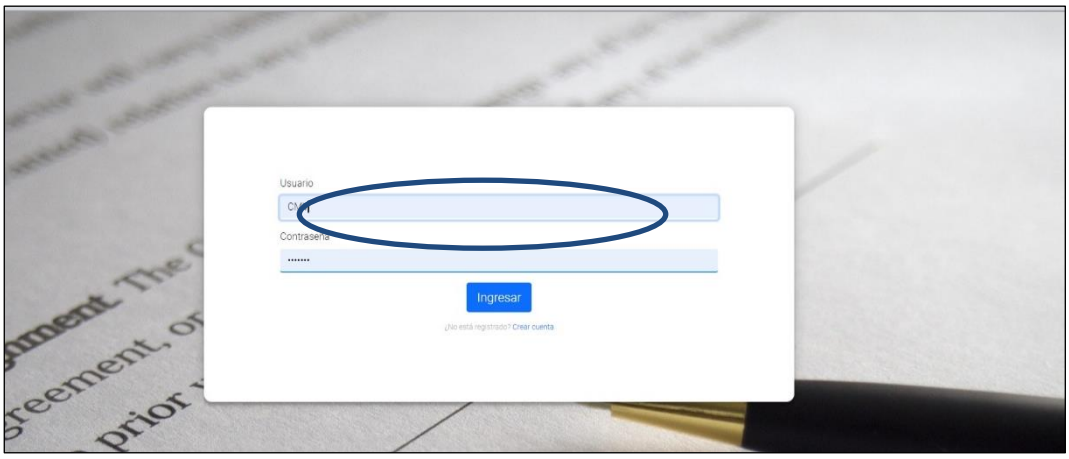


Figura: 28. Ventana de acceso

Una vez se accede, se muestra la interfaz inicial y un menú que muestra las funciones acordes a cada usuario, según el rol como actor (contratante, contratista o auditor) para la interacción con los contratos.

Figura: 29. Ventana de inicio

El contratante puede:

- Gestionar presupuestos (Crear y ver resumen junto con ejecución de presupuestos)
- Crear (construir y generar) contratos
- Aceptar (firmar o aprobar) contratos
- Editar contratos
- Revisar contratos y verificar resumen de contratos en ejecución
- Validar avances de cumplimiento del contrato y alertar incumplimientos
- Bloquear contratos

El contratista puede:

- Aceptar (firmar contratos)
- Visualizar contratos
- Reportar contratos (reportar al contratista la ejecución de las actividades que integran un contrato)

El auditor puede:

- Hacer seguimiento a las transacciones (investigación e historial de transacciones)

Presupuestos

El primer paso para que la plataforma genere transacciones y con ello ejecute smartcontracts en la blockchain consiste en que el contratante, registre los presupuestos (figura 30), que se constituyen en la disponibilidad de recursos financieros para la creación y ejecución de contratos.

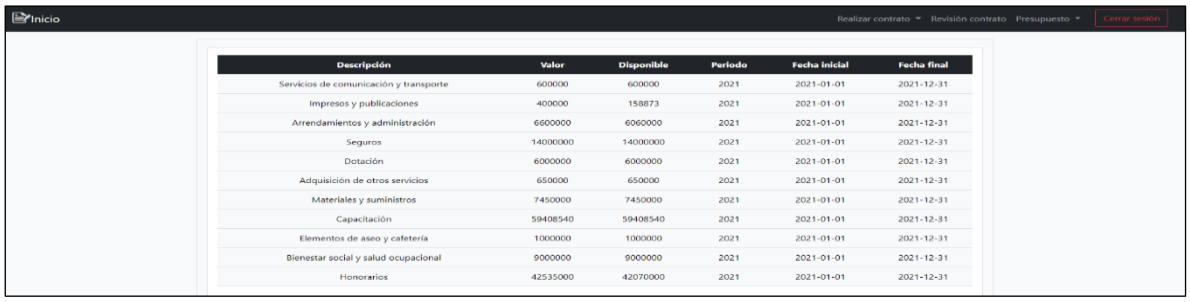
Se ingresan entonces los once (11) presupuestos que ha planificado la CMV en el PAA ejecutar a través de procesos de contratación pública. Cada presupuesto se ejecuta con uno o más contratos.

La plataforma permite ver cada presupuesto con su valor inicial adjudicado, así como su saldo o disponibilidad en la medida en que se van ejecutando y firmando transacciones relacionadas con reportes de seguimiento y ejecución de contratos. Esta información se actualiza automáticamente cuando se cuenta con el consenso entre contratistas y contratantes.

La plataforma no permite que se ejecuten mayores recursos a los disponibles, así como tampoco permite que los contratos se ejecuten en fechas que no se encuentren en el rango de fechas inicial y final de vigencia del presupuesto (figura 30). Con ello se evita, por un lado, la sobre-ejecución de recursos y por el otro, se mitiga la posibilidad de comprometer recursos

en fechas que, por principios presupuestales, no se podrían desembolsar so pena de faltas de orden disciplinario y fiscal.

Al ejecutar las pruebas se validó la condición de crear contratos si y solo sí, existen presupuestos con recursos disponibles, de lo contrario, no será posible dar incio al proceso.



Descripción	Valor	Disponible	Periodo	Fecha inicial	Fecha final
Servicios de comunicación y transporte	600000	600000	2021	2021-01-01	2021-12-31
Impresos y publicaciones	400000	158873	2021	2021-01-01	2021-12-31
Arrendamientos y administración	6600000	6060000	2021	2021-01-01	2021-12-31
Seguros	14000000	14000000	2021	2021-01-01	2021-12-31
Dotación	6000000	6000000	2021	2021-01-01	2021-12-31
Adquisición de otros servicios	650000	650000	2021	2021-01-01	2021-12-31
Materiales y suministros	7450000	7450000	2021	2021-01-01	2021-12-31
Capacitación	59408540	59408540	2021	2021-01-01	2021-12-31
Elementos de aseo y cafetería	1000000	1000000	2021	2021-01-01	2021-12-31
Bienestar social y salud ocupacional	9000000	9000000	2021	2021-01-01	2021-12-31
Honorarios	42535000	42070000	2021	2021-01-01	2021-12-31

Figura: 30. Ventana presupuestos

Contratos

El contratista puede crear contratos con cargo de recursos a los presupuestos del PAA y bajo una estructura mínima que permitirá los reportes y seguimientos posteriores, esta estructura contiene:

- Nombre del contrato y descripción u objeto contractual.

- Valor del contrato y asociación del presupuesto del cual se hará uso de los recursos. En este punto, para el caso de la aplicación del prototipo en la CMV, cada contrato tiene una sola fuente de financiación, es decir, los recursos provienen de un solo presupuesto. Sin embargo, en otras entidades públicas que manejan contratos con cuantías notablemente mayores, pueden cargar un solo proceso de contratación a más de un presupuesto y en este caso, se deberá establecer la proporción de recursos que aporta cada presupuesto para la ejecución del contrato.

- Fecha de inicio y finalización (no puede excederse del rango de fechas para las que es válido el presupuesto al cual se carga el contrato y en caso de suceder, la plataforma no permite dar continuidad a la transacción y genera la advertencia)

- Contratista que efectuará la suscripción y ejecución del contrato.

- Cronograma o conjunto de actividades o entregas parciales junto con el valor que desembolsará la entidad pública con el cumplimiento de cada una de ellas, hasta la entrega final y con ello, la ejecución total del valor acordado en el contrato. Se puede registrar una sola actividad y con ello un solo pago. Para el caso de actividades recurrentes se pueden incorporar sin problema, solamente que, para ser identificables, deben recibir un nombre diferente cada una de ellas. En este punto, un exceso de recursos en la programación de las actividades o fechas que no son coherentes con la vigencia del contrato, de inmediato generan una notificación indicando que la operación no puede ejecutarse y se pide la verificación y corrección de la información.

Como uno de los principios de la blockchain es la desintermediación, para que el contrato creado se formalice debe contar con la firma de las partes (contratista y contratante), es decir

que mientras el contrato esté firmado parcialmente por uno de los actores, la plataforma no permite efectuar reportes debido a que, bajo esa condición, no existe ningún acuerdo entre las partes. El contratista y el contratante tienen posibilidad de ver el contrato y luego de su visualización, deberá optar por firmarlo o rechazarlo.

Una vez se incluyó la totalidad de la información enunciada en los párrafos anteriores, los contratantes y contratistas al acceder, tendrán visualización del contrato suscrito como se evidencia en la figura 31.

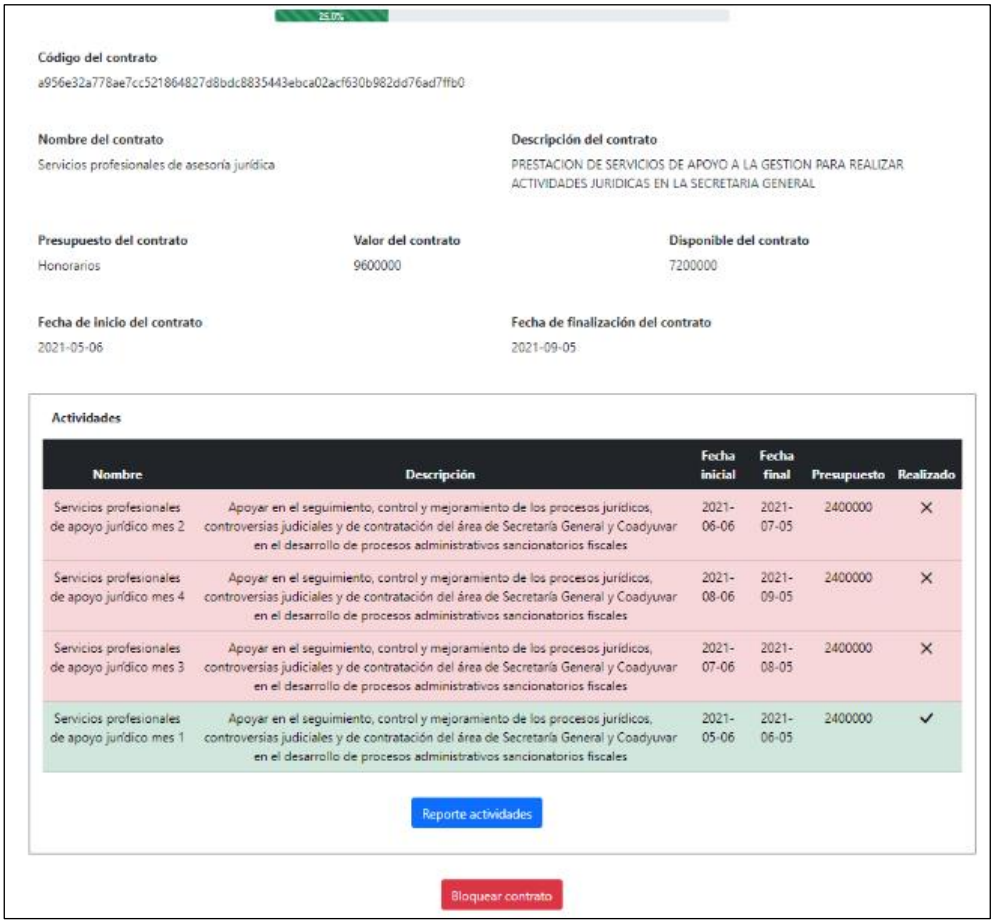


Figura: 31. Ventana gestión de contratos

Como uno de los principios de la blockchain es la desintermediación, para que el contrato creado se formalice debe contar con la firma de las partes (contratista y contratante), es decir que mientras el contrato esté firmado parcialmente por uno de los actores, la plataforma no permite efectuar reportes debido a que, bajo esa condición, no existe ningún acuerdo entre las partes. El contratista y el contratante tienen posibilidad de ver el contrato y luego de su visualización, deberá optar por firmarlo o rechazarlo.

Cualquier punto de un un contrato puede ser corregido antes de ser aceptado (firmado), después de esta acción hará parte de la blockchain y será inmodificable. En caso de que un contrato ya firmado que deba modificarse, se crea uno nuevo con los cambios necesarios y el tratamiento se dará como un nuevo contrato.

La plataforma permite reportar las actividades realizadas del contrato mediante la visualizión del detalle de cada una de esas actividades, su estado, presupuesto gastado y observaciones en caso de que el contratante lo considere necesario (figura 32). En este caso,

así como cuando se suscribió el contrato, de no existir consenso entre las partes, no se marcará el cumplimiento de la actividad, por tanto, no se aprobará el desembolso de recursos amparados en el contrato (figura 33).

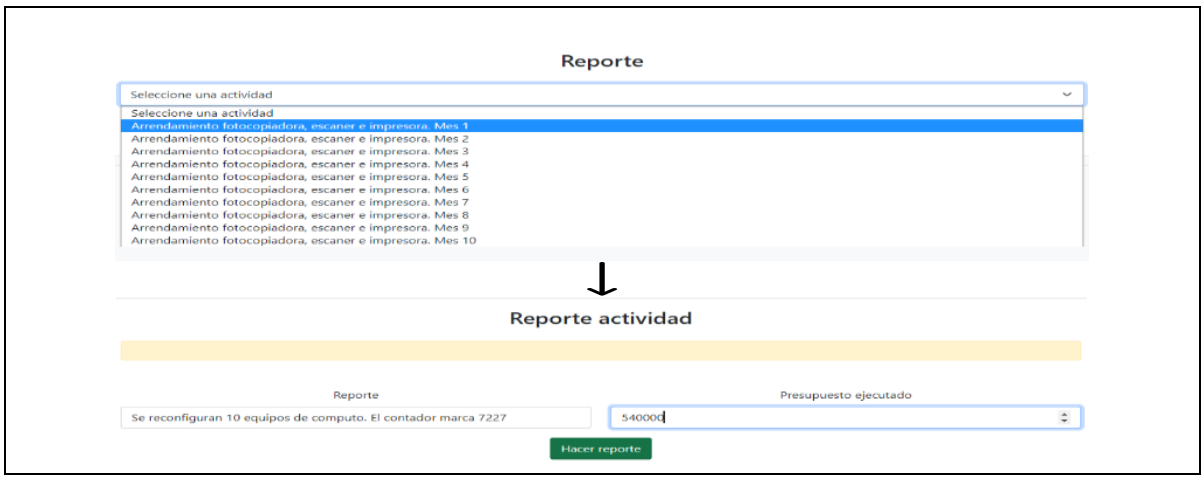


Figura: 32. Ventanas de reporte

Figura: 33. Ventanas de aprobación o rechazo de reporte

En la figura 31, la actividad que se encuentra en verde, se encuentra efectuada en su totalidad, mientras que las demás, al no contar con reportes de avances, se tienen como actividades con ejecución pendiente debido a que aún no han culminado las fechas de ejecución, establecidas en el contrato. Es importante enunciar que, en las pruebas, el contratista no puede efectuar un nuevo reporte hasta que se cuente con retroalimentación y autorización del contratante.

Asimismo, respecto del contrato, se puede visualizar el avance o progreso del cumplimiento del contrato con respecto al tiempo que tiene este para ser culminado, así como a las actividades acordadas (barra superior de la ventana de gestión de contratos).

Ahora bien, en caso de presentarse retrasos en tiempo o gastos fuera de los previstos, se genera una alerta de posibles incumplimientos tanto para el rol de contratante como para el rol de contratista. En este sentido, de ser necesario, se puede bloquear un contrato que esté siendo incumplido, para mitigar la posibilidad de materializar un riesgo de corrupción. En consecuencia, el contrato entra en proceso de investigación.

De manera permanente, con el rol de auditor se puede revisar el historial de las transacciones, esto es, de cada una de las actividades realizadas dentro de la plataforma registradas en el sistema blockchain, para hacer un seguimiento de un contrato, delimitación de presupuesto o de un usuario.

Para lo anterior, el auditor al ingresar a la revisión de transacciones selecciona los parámetros de búsqueda y revisa la información que requiera. Adicionalmente se puede generar un archivo en PDF para conservar el registro de la auditoría o realizar análisis posteriores. En el *Anexo 2* se presenta el archivo resultante de una auditoría.

Frente al potencial de la Blockchain para reforzar la trazabilidad y transparencia en la contratación pública, el prototipo ha evidenciado capacidad de integrar los requerimientos de roles y funcionalidades, asociados a la firma o aprobación de un contrato, gestión de su ejecución y asignación de recursos adecuados y bajo condiciones de oportunidad y la auditoría correspondiente e insertarlos en bloques encriptados de información de la red P2P.

El prototipo ha logrado vincular las características de los contratos desde su aprobación o firma hasta su culminación; con los presupuestos establecidos en el plan anual de adquisiciones y otorga así, al registro, atributos del blockchain: confianza en la inmutabilidad de los datos allí registrados, seguridad, trazabilidad y transparencia en la ejecución de las actividades de cada contrato con su respectivo pago a cargo del presupuesto destinado para ello, en un modelo descentralizado.

En cuanto al Smartcontract, se desarrolló una aplicación propia, con exclusiva funcionalidad en el método de inserción y consulta para el prototipo. El Smart contract aquí sirve para regular de manera automática la actividad asociada a la ejecución de las actividades de los contratos y las órdenes se ejecutan a medida que los requerimientos de cumplimiento y pago desde los presupuestos, se van cumpliendo.

2. Validación de funcionalidad

Se verificó la información de la contratación correspondiente al primer semestre de 2021 (utilizando el prototipo) comparada con el mismo periodo de 2020 (sin prototipo).

Los datos permitieron verificar condiciones de transparencia, coherencia y presunción de materialización de actos de corrupción para dos situaciones: sin el uso del prototipo (primer semestre de 2020) y con su implementación (primer semestre de 2021).

	Presupuesto anual	Número de contratos que se estima celebrar durante la anualidad	Celebrados primer semestre	Presupuesto comprometido primer semestre	
PAA 2020	$ 160.916.595	25	13	$ 89.527.233	56%
PAA 2021	$ 147.643.540	19	13	$ 69.622.721	47%

Tabla 10: Presupuestos y proyección de contratación en la CMV -2020 y 2021

Es importante indicar que para el caso del comportamiento de la contratación del primer semestre de 2020 se acudió a la información alojada en la plataforma SECOP, en la cual las entidades que contratan con cargo a recursos públicos publican los documentos de cada proceso. SECOP es una plataforma exclusivamente de publicidad, no es transaccional y es administrada por la Agencia Nacional de Contratación Pública.

	Semestre 1-2020	Semestre 1-2021
Hallazgo 1	Un contrato se dio por terminado de manera anormal después de ser convocado, debido a que el contratista no ofreció las especificaciones técnicas requeridas para su realización. La información que reposa en SECOP indica (luego de una revisión exhaustiva de los documentos) que el proceso fue declarado desierto, por tanto, el presupuesto de este contrato ($17.180.748), no se ejecutó (por lo menos no, durante el primer semestre de 2020). Por su parte, en la revisión de información relacionada con la contratación histórica que la CMV debe publicar en el marco de la política de *Transparencia y acceso a la información pública*, no permite visualizar que el contrato fue cancelado después de convocado y que no hubo ejecución presupuestal, en el informe de gestión correspondiente a ese semestre, se ve como ejecutado, es decir, con la forma de presentar información se lee que el valor del contrato se entregó al contratista sin existir vinculo contratual y sin demostrar el cumplimiento de los productos pactados. Presupuesto con observaciones: **$17.180.748**	Presenta claridad en la aprobación y ejecución de contratos
Hallazgo 2	Uno de los contratos planteaba la adquisición de elementos de protección personal (EPP) y en realidad se adquirió la dotación para los funcionarios que por Ley deben recibirla. Lo anterior significa que: se dio aval de cumplimiento del contrato por un producto que no se había acordado (o de manera informal se acordó la entrega de elementos de dotación, aunque en el contrato indicara que se estaban adquiriendo los EPP). Presupuesto con observaciones: **$3.910.000**	Presenta coherencia en lo contratado y lo entregado o ejecutado.
Hallazgo 3	Se encontró un contrato con presupuesto ejecutado de manera extemporánea (un mes después de lo pactado). Consultando SECOP, no se sabe por qué razón debido a que los documentos que soportan el cumplimiento oportuno de los compromisos por parte del contratista. Se indaga en la CMV y allí se indica que la entidad no contaba con presupuesto para cumplir con el pago acordado, por lo que se tuvo que esperar al año siguiente para pagar con cargo al presupuesto de 2021. Por otro lado, si el presupuesto del contrato estaba comprometido, no se pudo detectar la razón por la que, en el momento de pagar, no contaba con recursos. Presupuesto con observaciones: **$3.997.725**	Totalidad de pagos realizados oportunamente, previa validación del cumplimiento de las actividades o productos pactados en el contrato.
Hallazgo 4	Se recibió una denuncia acerca de un posible hecho de corrupción por concepto de contratación de capacitaciones por valor total de $22.500.000. La denuncia argumenta precios altos por hora de capacitación, sin embargo, ese atributo excede el alcance del prototipo. De todad formas, se verifica en en SECOP y se compara con el PAA y no se encuentra que el presupuesto se haya asignado al inicio de la anualidad para ese propósito (capacitaciones). Presupuesto con observaciones: **$22.500.000**	No se encuentra registro de denuncias asociadas a la contratación de la CMV. La totalidad de contratos se han suscrito con cargo a uno o más presupuestos contemplados en el PAA

Tabla 11: Resultados de la aplicación del prototipo en la CMV -Semestre 1 -2021

Para el análisis realizado con la aplicación del prototipo, se puede dar cuenta que del ingreso de los trece (13) contratos con fecha de inicio durante el primer semestre de 2021, no se detectó ninguna situación que ponga de manifiesto algún hecho de corrupción o de disponibilidad selectiva de información.

Para la misma cantidad de contratos efectuados durante el primer semestre de 2020, es decir, antes de implementar el prototipo y se destacan los cuatro (4) hallazgos que se presentan en la tabla anterior.

Con lo anterior, se deduce que el 53% del presupuesto comprometido en contratación efectuada durante el primer semestre de 2020, esto es, $47.588.473; se encuentran con cuestionamientos en términos de falta de transparencia o de posible materialización de hechos de corrupción.

Para 2021, de los trece (13) contratos con fecha de inicio a lo largo del primer semestre, se encuentran finalizados siete (7) y los sesis (6) restantes, se encentran aún en periodo de ejecución, sin embargo, cinco (5) contratistas para igual número de contratos ya han recibido pagos parciales con base en lo acordado entre las partes. Ninguno ha requerido aclaración, no se han recibido notifiaciones de posibles hechos de corrupción y el nivel de ejecución presupuestal, coincide con las bases de datos de contratación. La contratación del primer semestre ya se efectuó en su totalidad y la contratación para el segundo semestre se estima desde julio, con base en lo establecido en el PAA.

Es importante destacar que, así como se enunció en párrafos anteriores, que un hallazgo importante con la aplicación del prototipo en la CMV frente al análisis con otras entidades públicas, consistió en que de manera excepcional algunas de ellas asignan recursos de más de un presupuesto para la ejecución de un solo contrato. En ese sentido, con el estudio del Estatuto General de Contratación en la Administración Pública (Decreto Único Reglamentario 1082, 2015), el prototipo es aplicable a cualquier entidad pública en Colombia, no obstante, deberán ser incorporados otros smartcontracts, que dependerán de la cuantía de los contratos, cuyos requisitos presentan algunas diferencias según sean los rangos que establece el Estatuto.

BLOQUE IV
Conclusiones y trabajo futuro

ÍNDICE DEL BLOQUE

CAPÍTULO 11. CONCLUSIONES Y TRABAJO FUTURO

Una vez presentada la solución que esta tesis doctoral propone para el problema planteado y el marco teórico necesario para su realización, este capítulo abordará el cumplimiento de los objetivos trazados junto con la validez de la hipótesis, así como las conclusiones y el trabajo futuro.

1. Verificación de objetivos e hipótesis

En las secciones 1.2 y 1.3 de esta memoria se plantearon objetivos con el propósito de validar la hipótesis planteada, los mismos fueron contrastados evidenciando que han sido cumplidos durante el desarrollo de esta tesis doctoral.

A continuación, se muestra el objetivo principal:

> Diseñar un sistema fiable que por medio de Blockchain público, muestre los Smart Contracts de forma que no se pueda alterar y se mejoren los índices de transparencia de la administración pública colombiana en favor de la disminución de la corrupción.

Este objetivo se dividió en varios objetivos específicos para poder abordarlo de una forma más sencilla. A continuación, se mencionará cada uno, comentando su verificación, evaluación y grado de cumplimiento:

> **1.** Elaborar el estado del arte de Internet de las Cosas, Blockchain y los Smart Contracts.

Este objetivo se ha verificado mediante el desarrollo un ejercicio de revisión de publicaciones relacionadas con blockchain, smartcontracts y sus aplicaciones.

> **2.** Especificar las herramientas tecnológicas aplicables y los requerimientos para la propuesta de un sistema o meta-modelo de integración Blockchain y Smart Contracts.

A partir de los resultados obtenidos del primer objetivo se logró determinar que tipo de blockchain pública permitía compatiblidad para integración con smart contracts. A la vez que se efectua un análisis de casos de aplicación de blockchain con integración de smart contracts, en los sectores industrial y de servicios, se delimitan los requerimientos tanto técnicos como funcionales para el caso que da origen a esta tesis.

3. Analizar, diseñar, desarrollar e implementar un prototipo de sistema, modelos y meta-modelo de integración Blockchain y Smart Contracts

Una vez identificadas las herramientas tecnológicas y la identificación de requerimientos, se procedió al diseño, desarrollo e implementación del prototipo y se obtuvo el meta-modelo de integración de blockchain y smartcontracts para su aplicación en procesos de contratación.

4. Proponer y aplicar pruebas de validación para la propuesta.

Para la validación de la propuesta se realizaron pruebas en la contratación estatal de la Contraloría Municipal de Villavicencio, permitiendo la evaluación en real del prototipo.

A partir de la evaluación realizada, se pudo concluir que el prototipo es válido para establecer condiciones de transparencia y disminuir los niveles de corrupción, basados en un análisis comparativo realizado frente a contratación histórica de la misma entidad.

Con lo anterior, se puede afirmar que el meta-modelo y el prototipo dan cuenta de un sistema fiable que por medio de Blockchain público, integrando smartcontracts permita gestionar la contratación pública de manera que las transacciones que en ella se encuentran, no se puedan alterar y se mejoren los índices de transparencia de la administración pública colombiana en favor de la disminución de la corrupción, frente a situaciones diferentes a las ocasionadas por el código de conducta de los contratantes o contratistas ya que este último es un factor ajeno a la gobernanza de cualquier sistema tecnológico.

Por tanto, con la presentación del meta-modelo y del prototipo como solución, se da cumplimiento a los objetivos específicos tercero y cuarto, y también se verifica la siguiente hipótesis y respectiva pregunta de esta tesis doctoral:

La integración de Internet de las cosas y el Blockchain, permite reconocer la importancia de los contratos inteligentes en el desarrollo de aplicaciones en áreas asociadas al Estado (también denominado gobierno o sector público) en su lucha contra la corrupción, debido a su versatilidad, seguridad, acceso y control del trámite o proceso que se esté realizando; dado que se cuenta con un sistema de información permanente y público.

¿Es posible la integración de blockchain y los smart contracts a través de un meta-modelo aplicado al sector público?

Como consecuencia de los resultados alcanzados en la investigación se puede inferir que la misma ha sido validada como positiva.

2. Conclusiones

Con el correcto diseño y aplicación de un sistema que se beneficie del uso de blockchain público con smart contracts, que sea fiable, inalterable y seguro; se pueden mejorar los índices de transparencia de la administración pública colombiana en favor de la disminución de la corrupción, toda vez que, si alguien realiza un cambio, se verá como las partidas presupuestales cambian, debido a que con blockchain los registros son inalterables y con la implementación de smartcontracts, las etapas de contratación y de ejecución presupuestal en las que hay mayor susceptibilidad de materialización de riesgos de corrupción, serán autoejecutables una vez se cumplan los acuerdos o reglas establecidos previamente por las partes, de manera que no hay intervención de terceros que validen el cumplimiento de las condiciones contractuales, por lo que cada modificación será evidente y transparente.

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

Actualmente, las plataformas disponibles en el sector público colombiano son en últimas, repositorios documentales y no son transaccionales. De manera que una vez se va generando la evidencia de firma, ejecución, desembolso y liquidación de contratos, un tercero, además, humano, ingresa la información respectiva, esto en un plazo máximo que según la plataforma en la que se deba publicar la información, oscila entre los tres (3) y los cinco (5) días hábiles posteriores a su generación.

Cabe señalar que previamente o incluso, de manera paralela, los soportes asociados a la elaboración, seguimiento y cierre (liquidación o cancelación) del contrato, se deben radicar o deben obtener un registro de ingreso o de autorización por parte de la entidad contratante, donde la documentación es verificada y en los casos que no cumpla con los requisitos mínimos, será devuelta a los contratistas para subsanar lo que corresponda. Por lo que, en este punto, el contrato independiente de la etapa en la que se encuentre, pudo tener manipulación en su contenido o estructura debido a la cantidad de intermediarios a su paso hasta la aprobación o firma por parte del contrante, sin que exista mayor trazabilidad.

En cuanto a la interacción con la plataforma de contratación del Estado Colombiano, se hace hincapié en la palabra “humano” cuando se señala que la documentación se debe publicar allí, debido a que en estos casos hay mayor lugar a presentarse errores en el cargue o predisposición a alterar la información en favor propio o de terceros.

Por otro lado, únicamente esta persona tendrá acceso a la trazabilidad del proceso. De manera que no hay ejecución automática, carente de precisión y en algunos casos, de oportunidad, en donde la información está depositada en manos de un operador

tecnológico, situación que difiere de la funcionalidad del prototipo donde los datos en blockchain están distribuidos en la red.

Con la solución planteada, estos reprocesos, demoras, falta de trazabilidad y de transparencia y con ello, la posibilidad de configurarse hechos de corrupción, presentan reducción, debido a las ventajas suministradas por blockchain y los smartcontracts como se ha mencionado en el *Capítulo 10: Validación, pruebas y resultados*.

Por otro lado, en Colombia con la expedición del Plan Nacional de Desarrollo, se da un importante salto a la modernización del Estado, toda vez que en los artículos 147 y 148 del plan enunciado establece la *"Transformación digital pública"* y la *"Política de Gobierno Digital",* dando vía libre a la incorporación de componentes asociados a tecnologías emergentes y a la Cuarta Revolución Industrial en las entidades del estado del orden nacional, en búsqueda del incremento de la confianza y la seguridad digital y el fomento a la participación y la democracia por medios digitales; incluyéndose dentro de esas tecnologías la blockchain.

A pesar de la existencia de un Plan Nacional de Desarrollo que le otorga relevancia a la transparencia, a la confianza, a la seguridad digital y con ello a la lucha contra la corrupción, Colombia aún no cuenta con reglamentación que permita la implementación extensa y formal de plataformas basadas en tecnologías emergentes y para el caso del prototipo presentado, no precisamente por limitaciones en la normativa asociada al uso de blockchain, sino a temas de contratación por medios no convencionales, protección de datos, seguridad de la información, identidad digital.

En consecuencia, se requiere de un marco legal claro, libre de ambigüedades o de vacíos, de ser posible, basado en las mejores prácticas internacionales, para una excelente gestión de la contratación electrónica, que responda a las necesidades del país en su lucha contra la corrupción, así como a los requerimientos de organismos y autoridades internacionales en la materia.

Lo anterior, claramente limita por ahora, la puesta en producción del prototipo a nivel no de entidades, sino del Estado Colombiano. A nivel de entidades puede aplicarse a nivel de pruebas piloto y de buenas prácticas para mejorar la gestión de cada entidad y sin que ello reemplace los demás trámites tradicionales que el estatuto de contratación exige.

Otro punto para destacar consiste en que la alicación de este prototipo a nivel de las entidades públicas, de manera complementaría al propósito inicial establecido, consiste en el cumplimiento gradual de la política de Datos Abiertos del estado colombiano, que tiene como propósito, establecer el desarrollo de estrategias de apertura y reúso de datos, que estén orientadas a la generación de transparencia, control social, valor público, económico, académico, cultural, ambiental, y en general, en los distintos ámbitos de la sociedad.

Para el caso de la aplicación específica en materia de la gestión de contratos suscritos con Entidades Públicas, se pudo concluir que el prototipo aplicado es válido para establecer condiciones de transparencia, para reducir denunciar por presuntos hechos de corrupción y para disminuir contratación histórica de la misma entidad (austeridad).

Adicionalmente, se puede afirmar que este desarrollo aporta valor en la transferencia de conocimiento, al aprovechar el prototipo actual y su implementación en la Contraloría Municipal de Villavicencio, para la creación de contratos nuevos, mejorar los existentes y escalarlos con un enfoque funcional que pueda fortalecer la posterior sostenibilidad, interoperabilidad y escalabilidad del modelo, pero también con un enfoque orientado al cumplimiento legal.

3. Trabajo futuro

Una vez llegado a este punto, ya se ha expuesto la forma cómo los objetivos de esta tesis se han cumplido, pero aún así queda mucho trabajo por desarrollar en el marco de tecnologías emergentes para el sector público.

A continuación, se recopilan algunas ideas de trabajo futuro que puede realizarse a partir de esta tesis doctoral:

Los ***smarts contracts*** en este caso fueron incorporados en blockchain para ser aplicados en un caso de procesos de contratación estatal, no obstante, se les puede sacar provecho en otros campos del sector público y privado, sea en sectores económicos de bienes o de servicios y en temas de la misionalidad de cada organización o de soporte (lo que se denomina en ocasiones, procesos administrativos), impactando la arquitectura corporativa y la gobernanza empresarial, gracias a la automatización y a la desintermediación que brindan los ***smart contracts*** en conjunto con ***blockchain***, ya que ofrecen gran nivel de descentralización, inmutabilidad y transparencia por lo que restarían pocas posibilidades de ocultar acciones corruptas o negar responsabilidades.

En seguida algunos casos en concreto:

- *Licitaciones:* Puede utilizarse blockchain para certificar los procesos de compras públicas y mantener visible la trazabilidad desde la licitación hasta la compra final.
- *Gestión interna de la organización:* Permitiría coordinación de las personas y la distribución de roles y responsabilidades, así como de los flujos de trabajo.
- *Contabilidad:* Los compromisos económicos de una organización, con sus partes interesadas, por ejemplo, proveedores, clientes, accionistas; si se definen claramente por reglas autoejecutables mediante smartcontracts, deja poco espacio para las interpretaciones y para ambigüedades. Las personas que autorizan cada gasto del presupuesto serían las responsables de manera automática es indiscutible.
- *Participación ciudadana en la toma de decisiones:* Madurar las actuales plataformas orientadas a la digitalización (automatizar y optimizar procesos), a plataformas que también sean transformadoras, es decir de participación incidente por parte de la ciudadanía, la empresa y la sociedad civil; en términos de plantear demandas y necesidades y ofrecer mejoras y soluciones, en casos como servicios públicos, servicios sociales, destinación de presupuestos, aprobación y seguiento a los planes de desarrollo o planes sectoriales.

Adicionalmente, se puede investigar acerca del uso de block chain y Smart contracts, incorporando ***Inteligencia Artificial (IA)*** para agilizar o automatizar los procesos

públicos y mejorar los servicios que se prestan a la ciudadanía, haciendo los servicios gubernamentales más personalizados, disponibles y oportunos, generando en ellos mayor cercanía, confianza y transparencia. A la vez, con la implementación de soluciones tecnológicas basadas en IA, para los procesos de las entidades públicas se podría tener alivio operativo logrando que la tecnología se encargue de aliviar las labores rutinarias de los funcionarios públicos de manera que estos puedan enfocarse en temas más profundos que atiendan las problemáticas de cada sector y contribuyan en el cumplimiento de su quehacer o misionalidad.

Por otro lado, se debe reconocer que la operación del sector público se ha caracterizado por su burocracia y lentitud; no obstante, con ocasión de la pandemia generada por la COVID-19, se dio origen a una creciente digitalización de las interacciones, procedimientos o trámites públicos, que generalmente se basan en datos y que también generan más datos. Ello sumado a los datos abiertos que existen disponibles en el ecosistema digital, pone a los gobiernos frente al uso, valoración y custodia de estos activos de información; por lo que resulta importante analizar la posibilidad de recurrir a tecnologías basadas en ***IA*** y en ***analítica de datos*** para procesar las grandes cantidades de datos y para poder tomar decisiones.

CAPÍTULO 12. PUBLICACIONES DERIVADAS

Durante la realización del doctorado se han escrito diferentes publicaciones para difundir ante la comunidad científica. A continuación, se enumeran las publicaciones, en orden cronológico:

- Rodríguez-Molano J.I., Triana-Casallas J.A., Contreras-Bravo L.E. (2018) ***Modeling and Simulation of Integration of Internet of Things and Manufacturing Industry 4.0.*** In: Figueroa-García J., Villegas J., Orozco-Arroyave J., Maya Duque P. (eds) Applied Computer Sciences in Engineering. WEA 2018. Communications in Computer and Information Science, vol 916. Springer, Cham. https://doi.org/10.1007/978-3-030-00353-1_21

- Rodríguez Molano, J. I., Martínez Baracaldo, J. N., & Triana Casallas, J. A. (2020). ***Prospective for the integration of Blockchain and the IoT for Cluster implementation***. *Ingeniería Solidaria*, *16*(3), 1-30. https://doi.org/10.16925/2357-6014.2020.03.06

- Triana, Jenny & Cueva-Lovelle, Juan & Rodríguez Molano, José. (2020). ***Smart Contracts with Blockchain in the Public Sector***. International Journal of Interactive Multimedia and Artificial Intelligence. In Press. 10. 10.9781/ijimai.2020.07.005.

- Triana Casallas, J., Rodríguez Molano J., Fuentes Hector J. Artículo: ***"MEVF a DSL proposed for interoperability and management of fiscal scenarios"***. Publicado en: International Journal of Mechanical and Production Engineering Research and Development (IJMPERD) ISSN (P): 2249–6890; ISSN (E): 2249–8001 Vol. 10, Issue 6, Dec 2020, 491–498

- Triana, J. A., Rodriguez, J. I., & Contreras, L. E. (2020). ***Run-Time Optomization using the invokedynamic statement***. Publicado en: International Journal of Mechanical and Production Engineering Research and Development (IJMPERD), ISSN (P): 2249–6890; ISSN (E): 2249–8001. Vol. 10(6), 517–524.

CAPÍTULO 13. BIBLIOGRAFÍA

Al-Batran, B., Schätz, B., & Hummel, B. (2012). Model Driven Engineering Languages and Systems. *MoDELS*, *7590*(OCTOBER), 258–272. https://doi.org/10.1007/978-3-642-33666-9

Alharby, M., & van Moorsel, A. (2017). BLOCKCHAIN-BASED SMART CONTRACTS : A SYSTEMATIC MAPPING STUDY. *Computer Science & Information Technology (CS & IT)*, 125–140. https://doi.org/10.5121/csit.2017.71011

Ali, M. S., Vecchio, M., Pincheira, M., Dolui, K., Antonelli, F., & Rehmani, M. H. (2018). Applications of Blockchains in the Internet of Things: A Comprehensive Survey. *IEEE Communications Surveys and Tutorials*, *PP*(X), 1. https://doi.org/10.1109/COMST.2018.2886932

Andreas Antonopoulos. (2016). *Reflexiones sobre el futuro del dinero*.

Angel, M., Zambrano, N., Cobos, C., & Mendoza, M. E. (2004). *Unicauca Virtual: Metamodelos de Universidad Virtual y Herramientas de Soporte*. https://www.researchgate.net/publication/251776449

Antonio, L., Rubio, S., & Huertas, A. B. (2018). *Blockchain: La revolución de la confianza digital (spanish)*. https://www.sic.gov.co/boletines-tecnologicos/blockchain-la-revolucion-de-la-confianza-digital

Ardila, C. Q. (2018). *La naturaleza económica del Bitcoin: un enfoque monetario (spanish)*. https://www.mail-archive.com/cryptography@metzdowd.com/msg09959.html

Armas, M. (2012). *Model Driven Architecture (MDA)*. https://sg.com.mx/content/view/861

Atkinson, C., & Kühne, T. (2007). A Tour of Language Customization Concepts. *Advances in Computers*, *70*, 105–161. https://doi.org/10.1016/S0065-2458(06)70003-1

B. Selic. (2008). MDA Manifestations. *The European Journal for the Informatics Profesional*, 12–16.

Balasubramanian, K., Gokhale, A., Karsai, G., Sztipanovits, J., & Neema, S. (2006). Developing applications using model-driven design environments. *Computer*, *39*(2), 33–40. https://doi.org/10.1109/MC.2006.54

Bambara, J. J. . P. R. A. K. I. R. M. S. L. M. W. (2018). *Blockchain: A Practical Guide to Developing Business, Law, and Technology solutions* (McGraw Hill

Professional (ed.)). https://books.google.com.co/books?id=z5hIDwAAQBAJ&hl=es

Bapty, T. A., & Sztipanovits, J. (1997). Model-based engineering of large-scale real-time systems. *Proceedings of the International Workshop on Engineering of Computer-Based Systems*, 467–474. https://doi.org/10.1109/ECBS.1997.581931

Baran Paul. (1964). On Distributed Communications. In *Computer Communications* (Vol. 2, Issue 3, pp. 137–138). https://doi.org/10.1016/0140-3664(79)90214-7

Bashir, I. (2017a). *Mastering Blockchain: Distributed ledger, decentralization, and smart contracts explained* (L. Subramanian (ed.)). Packt Publishing. https://books.google.com.sa/books?hl=ar&lr=&id=3ZlUDwAAQBAJ&oi=fnd&pg=PP1&dq=mastering+blockchain+distributed+ledger+technology+pdf&ots=-4n3EITOZH&sig=x2e-Mp8HWDqOG5AZkyCdH3689uU&redir_esc=y#v=onepage&q=mastering blockchain distributed ledger technology p

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

BBVA Research. (2015). *Situación económica digital*. www.bbvaresearch.com

Bézivin, J. (2005a). On the Unification Power of Models On the Unification Power of Models 1. *Software and Systems Modeling (SoSyM)*, *4*(2), 171–188. https://doi.org/10.1007/s10270-005-0079-0

Bézivin, J. (2005b). On the unification power of models. *Software & Systems Modeling 2005 4:2*, *4*(2), 171–188. https://doi.org/10.1007/S10270-005-0079-0

Casino, F., Dasaklis, T. K., & Patsakis, C. (2019). A systematic literature review of blockchain-based applications: Current status, classification and open issues. In *Telematics and Informatics* (Vol. 36, pp. 55–81). https://doi.org/10.1016/j.tele.2018.11.006

CEPAL-UN. (n.d.). *¿Qué es el gobierno abierto? - De Gobierno Abierto a Estado Abierto - Biblioguias at Biblioteca CEPAL, Naciones Unidas*. Retrieved May 16, 2021, from https://biblioguias.cepal.org/EstadoAbierto/concepto

Christidis, K., & Devetsikiotis, M. (2016). Blockchains and Smart Contracts for the Internet of Things. In *IEEE Access* (Vol. 4, pp. 2292–2303). https://doi.org/10.1109/ACCESS.2016.2566339

Chung, T.-Y., Mashal, I., Alsaryrah, O., Huy, V., Kuo, W.-H., & Agrawal, D. P. (2013). Social Web of Things: A Survey. *2013 International Conference on Parallel and Distributed Systems*, 570–575.

https://doi.org/10.1109/ICPADS.2013.102

Comité, A. L., Social, E. Y., & Al, E. Y. (2018). *Communication from the Commission to the European Parliament, The Council, The European Economic and Social Committee and the Committee of the Regions*. European Commission. https://ec.europa.eu/info/strategy/international-strategies/global-topics/sustainable-development-goals/eu-

Cong, L. W., & He, Z. (2018). *Blockchain Disruption and Smart Contracts*.

De Filippi, P. (2014). *Primavera De Filippi on Ethereum: Freenet or Skynet?* The Berkman Center for Internet and Society at Harvard University. https://www.youtube.com/watch?v=slhuidzccpI

De Ugarte, D. (2018). El Poder de las redes. In *Libro de grabados*. https://doi.org/10.2307/j.ctv86dgt3.11

Diaz, V. (2011). *MDCI : Model-Driven Continuous Integration*. Universidad de Oviedo.

Dolader, C. R., Roig, J. B., & Muñoz Tapia, J. L. (2017). La blockchain: Fundamentos, aplicaciones y relación con otras tecnologías disruptivas. *Economía Industrial*, *405*(Nuevas tecnologías digitales). https://www.mincotur.gob.es/Publicaciones/Publicacionesperiodicas/EconomiaIndustrial/RevistaEconomiaIndustrial/405/DOLADER, BEL Y MUÑOZ.pdf

Drescher, D. (2018). Blockchain basics: a non-technical introduction in 25 steps. *Financial Markets and Portfolio Management*, *32*(3), 329–331. https://doi.org/10.1007/s11408-018-0315-6

Fang, C., Liu, X., Pardalos, P. M., & Pei, J. (2016). Optimization for a three-stage production system in the Internet of Things: procurement, production and product recovery, and acquisition. *International Journal of Advanced Manufacturing Technology*, *83*(5–8), 689–710. https://doi.org/10.1007/s00170-015-7593-1

Fuentes, L., & Vallecillo, A. (2004). Una Introducción a los Perfiles UML. *Novatica*, *168*, 6–11.

Gama, K., Touseau, L., & Donsez, D. (2012). Combining heterogeneous service technologies for building an Internet of Things middleware. *Computer Communications*, *35*(4), 405–417. https://doi.org/10.1016/j.comcom.2011.11.003

García-Díaz, V. (2011). *MDCI: Model-driven continuous integration* [UNIVERSIDAD DE OVIEDO]. http://iospress.metapress.com/index/M6V716216H21J2X4.pdf

García-Díaz, Vicente, Fernández-Fernández, H., Palacios-González, E., G-Bustelo,

B. C. P., Sanjuán-Martínez, O., & Lovelle, J. M. C. (2010). TALISMAN MDE: Mixing MDE principles. *Journal of Systems and Software*, *83*(7), 1179–1191. https://doi.org/10.1016/J.JSS.2010.01.010

García-Díaz, Vicente, Tolosa, J. B., G-Bustelo, B. C. P., Palacios-González, E., Sanjuan-Martínez, Ó., & Crespo, R. G. (2009). TALISMAN MDE framework: An architecture for intelligent model-driven engineering. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *5518 LNCS*(PART 2), 299–306. https://doi.org/10.1007/978-3-642-02481-8_43

García, C. G., & Espada, J. P. (2013). Using Model-Driven Architecture Principles to Generate Applications based on Interconnecting Smart Objects and Sensors. In *Advances and Applications in Model-Driven Engineerin* (pp. 365–385). https://doi.org/10.4018/978-1-4666-6359-6.ch015

Garcia, C. G., Espada, J. P., Valdez, E. R. N., & Diaz, V. G. (2014). Midgar: Domain-Specific Language to Generate Smart Objects for an Internet of Things Platform. *2014 Eighth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, 352–357. https://doi.org/10.1109/IMIS.2014.48

García Mateo, P. (2018). *Blockchain aplicado al sector público (spanish)* [Universidad Politécnica de Valencia]. https://riunet.upv.es/handle/10251/111762

Giancaspro, M. (2017). Is a 'smart contract' really a smart idea? Insights from a legal perspective. *Computer Law and Security Review*, *33*(6), 825–835. https://doi.org/10.1016/j.clsr.2017.05.007

Gonzalez, C. (2017). *MIDGAR: interoperabilidad de objetos en el marco de Internet de las Cosas mediante el uso de Ingeniería Dirigida por Modelos*. Universidad de Oviedo.

Gronback, R. C. (2009). *Eclipse modeling project : a domain-specific language toolkit* (Pearson Education (ed.)). Addison-Wesley.

Gupta, M. (2017). *Blockchain for dummies*.

Hailpern, B., & Tarr, P. (2006). Model-driven development: The good, the bad, and the ugly. *IBM Systems Journal*, *45*(3), 451–461. https://doi.org/10.1147/sj.453.0451

Hardwick, F. S., Akram, R. N., & Markantonakis, K. (2018). Fair and Transparent Blockchain Based Tendering Framework - A Step Towards Open Governance. *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, 1342–1347.

https://doi.org/10.1109/TrustCom/BigDataSE.2018.00185

Herrmannsdoerfer, M., Benz, S., & Juergens, E. (2009). COPE - Automating Coupled Evolution of Metamodels and Models. *ECOOP 2009 - Object-Oriented Programming*, *July*, 52–76. https://doi.org/10.1007/978-3-642-03013-0

Hou, H. (2017). The application of blockchain technology in E-government in China. *2017 26th International Conference on Computer Communications and Networks, ICCCN 2017*, 1–4. https://doi.org/10.1109/ICCCN.2017.8038519

Jesús García Molina, Félix García, Vicente Pelechano, Antonio Vallecillo, Juan Manuel Vara, & Cristina Vicente-Chicote. (2014). *Desarrollo de Software Dirigido por Modelos: Vol. I*. Ra-Ma. http://www.lcc.uma.es/~av/Publicaciones/12/LibroDSDM.pdf

Juan Bernardo Quintero, & Anaya, R. (2007). MDA y el papel de los modelos en el proceso de desarrollo de software. *EIA*, *8*. http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S1794-12372007000200011

Kent, S. (2002). Model driven engineering. *International Conference on Integrated Formal Methods*, 286–298.

La blockchain negli appalti pubblici, come utilizzarla: i vantaggi | Agenda Digitale (italian). (n.d.). Retrieved May 29, 2020, from https://www.agendadigitale.eu/procurement/la-blockchain-negli-appalti-pubblici-come-utilizzarla-i-vantaggi/

Lucas, M. (2019). *Tecnología blockchain. Un nuevo modelo de acción de gobierno - ACOP*. A Fondo. https://compolitica.com/tecnologia-blockchain-un-nuevo-modelo-de-accion-de-gobierno/

Luu, L., Chu, D.-H., Olickel, H., Saxena, P., & Hobor, A. (2016). Making Smart Contracts Smarter. *Conference on Computer and Communications Security*, 254–269. https://doi.org/10.1145/2976749.2978309

Macrinici, D., Cartofeanu, C., & Gao, S. (2018). Smart contract applications within blockchain technology: A systematic mapping study. In *Telematics and Informatics* (Vol. 35, Issue 8, pp. 2337–2354). https://doi.org/10.1016/j.tele.2018.10.004

Maheshwari, S. (2018). *Blockchain basics: Hyperledger Fabric -IBM Developer*. https://developer.ibm.com/articles/blockchain-basics-hyperledger-fabric/?mhsrc=ibmsearch_a&mhq=hyperledger

Makhdoom, I., Abolhasan, M., Abbas, H., & Ni, W. (2019). Blockchain's adoption

in IoT: The challenges, and a way forward. *Journal of Network and Computer Applications*, *125*, 251–279. https://doi.org/10.1016/j.jnca.2018.10.019

Manuel, J., Lovelle, C., & Inform, S. (2019). *Internet de las cosas (spanish)*.

Martínez, S., Lamoth, L., Moreno, R., & Jacho, N. (2015). *Análisis de la Transformación de Modelo CIM a PIM en el Marco de Desarrollo de la Arquitectura Dirigida por Modelos (MDA)*. *36*(3).

Miller, J., Mukerji, J., & Belaunde France, M. (2003). *MDA Guide Version 1.0.1*.

Morabito, V. (2017). The Security of Blockchain Systems. In *Business Innovation Through Blockchain*. Springer International Publishing. https://doi.org/10.1007/978-3-319-48478-5_4

Moralejo, J. A. (2018). Blockchain en procesos de participación ciudadana (spanish). In Centro de Estudios Políticos y Constitucionales. Mº de la Presidencia (Ed.), *Participación ciudadana:experiencias inspiradoras en España* (pp. 147–158). http://www.gigapp.org/images/docus/Participacion ciudadana_11 Arteaga.pdf

Nakamoto, S. (2008). *Bitcoin: A Peer-to-Peer Electronic cash system*. Bitcoin. https://git.dhimmel.com/bitcoin-whitepaper/

Nathan, A. J., & Scobell, A. (2012). How China sees America. In *Foreign Affairs* (Vol. 91, Issue 5, pp. 1–30). https://doi.org/10.1017/CBO9781107415324.004

Núñez-Valdez, E. R., García-Díaz, V., Lovelle, J. M. C., Achaerandio, Y. S., & González-Crespo, R. (2016). A model-driven approach to generate and deploy videogames on multiple platforms. *Journal of Ambient Intelligence and Humanized Computing*, 1–13. https://doi.org/10.1007/s12652-016-0404-1

Ocariz Emiliano B. (2019). *Blockchain y smart contracts. La revolución de la confianza (spanish)* (Alfaomega (Ed.)). https://www.libreriadelau.com/blockchain-y-smart-contracts-la-revolucion-de-la-confianza-alfaomega-emprendimiento-e-innovacion/p

OMG, O. M. G. (2007). *MOF 2.0/XMI Mapping, Version 2.1.1*. 120. http://www.omg.org/spec/XMI/20071001

OMG, O. M. G. (2014). *OMG Document -- ormsc/14-06-01 (MDA Guide revision 2.0)*. https://www.omg.org/cgi-bin/doc?ormsc/14-06-01

OMG, O. M. G. (2019). *The architecture of choice for a changing world*. https://www.omg.org/mda/

Open Government Partnership. (2019). *Open Government Partnership Global*

Report: Democracy beyond the ballot box. https://doi.org/10.1145/2591888.2591924

Organisation for Economic Co-operation and Development (OECD). (2017). *OECD Recommendation of The Council on public integrity*. https://www.oecd.org/gov/ethics/recomendacion-sobre-integridad-es.pdf

Portmann, E. (2018). Rezension „Blockchain: Blueprint for a New Economy". *HMD Praxis Der Wirtschaftsinformatik*. https://doi.org/10.1365/s40702-018-00468-4

Decreto Único Reglamentario 1082, (2015). https://www.alcaldiabogota.gov.co/sisjur/normas/Norma1.jsp?i=71552

Preukschat Carlos Kuchkovsky, A., Gómez Lardies, G., Díez García Íñigo Molero, D., Luis Várez, J., Felguera, E., Steck, C., Madrid, I., Lage, Ó., Nespral, D., Díaz, R., Hamann, S., Fernández, C., Fernández, R., Junestrand, S., Contreras, A., Moreno, F., Vivas, C., Molina, J., Foz, X., … Polo Alex Puig, M. (2017). *Blockchain: la revolución industrial de internet (spanish)* (Gestión 2000 (Ed.)).

Prusty, N. (2017). *Building Blockchain Projects Develop real-time practical DApps using Ethereum and JavaScript*. Birmingham. https://learning.oreilly.com/library/view/building-blockchain-projects/9781787122147/

Redes centralizadas VS distribuidas. | by iCommunity.io | Medium. (n.d.). Retrieved May 12, 2021, from https://medium.com/@helloicommunity/redes-centralizadas-vs-distribuidas-2fc50c51f284

Reijers, W., O'Brolcháin, F., & Haynes, P. (2016). Governance in Blockchain Technologies & Social Contract Theories. *Ledger*, *1*, 134–151. https://doi.org/10.5195/ledger.2016.62

Reyna, A., Martín, C., Chen, J., Soler, E., & Díaz, M. (2018). On blockchain and its integration with IoT. Challenges and opportunities. *Future Generation Computer Systems*, *88*, 173–190. https://doi.org/10.1016/j.future.2018.05.046

Rodríguez Molano, J. I., Martínez Baracaldo, J. N., & Triana Casallas, J. A. (2020). Prospective for the integration of Blockchain and the IoT for Cluster implementation. *Ingeniería Solidaria*, *16*(3), 1–30. https://doi.org/10.16925/2357-6014.2020.03.06

Rosic, A. (2017). *What Are Smart Contracts? [Ultimate Beginner's Guide to Smart Contracts]*. https://blockgeeks.com/guides/smart-contracts/

Rumbaugh, J. (2005). *The unified modeling language reference manual* (I.

Jacobson & G. Booch (Eds.); 2nd ed.) [Book]. Addison-Wesley.

Savelyev, A. (2017). Contract law 2.0: 'Smart' contracts as the beginning of the end of classic contract law. *Information and Communications Technology Law*. https://doi.org/10.1080/13600834.2017.1301036

Schmidt, D. C. (2006). Guest Editor's Introduction: Model-Driven Engineering. *Computer*, *39*(2), 25–31. https://doi.org/10.1109/MC.2006.58

Selic, B. (2003). Model-driven development of real-time software using OMG standards. *ISORC 2003: SIXTH IEEE INTERNATIONAL SYMPOSIUM ON OBJECT-ORIENTED REAL-TIME DISTRIBUTED COMPUTING, PROCEEDINGS*, 4–6.

Selic, Bran. (2003a). Models, Software Models and UML [Inbook]. In L. Lavagno, G. Martin, & B. Selic (Eds.), *UML for Real: Design of Embedded Real-Time Systems* (pp. 1–16). Springer US. https://doi.org/10.1007/0-306-48738-1_1

Selic, Bran. (2003b). The pragmatics of model-driven development. *IEEE Software*, *20*(5), 19–25. https://doi.org/10.1109/MS.2003.1231146

Shermin, V. (2017). Disrupting governance with blockchains and smart contracts. *Strategic Change*. https://doi.org/10.1002/jsc.2150

Singh. (2017). *What is the Difference between Blockchain and Database?* https://atozmarkets.com/news/difference-between-blockchain-and-database/

Smith, C. (2011). *Blueprints for a new economy*. Nation. https://www.thenation.com/article/archive/exchange-blueprints-new-economy/

Stahl, T., Völter, M., Bettin, J., Haase, A., & Helsen, S. (2013). *Model-driven software development: technology, engineering, management*. John Wiley & Sons.

Stahl, Thomas, Völter, M., Bettin, J., Haase, A., & Helsen, S. (2013). *Model-Driven Software Development: Technology, Engineering, Management*. 446.

Stefanescu, D. I. (2019). *Blockchain – estudio de alternativas e implementaciones* [Universidad del País Vasco]. https://lsi.vc.ehu.eus/pablogn/docencia/PFC/Memoria TFG - Denis Ionut Stefanescu.pdf

Steinberg, D., Budinsky, F., Paternostro, M., & Merks, E. (2009). *Eclipse Modeling Framework*. Addison-Wesley.

Suárez Alvaro. (n.d.). *Blockchain: pros y contras*. Retrieved May 17, 2021, from https://medium.com/@alvarosb999/1-2-blockchain-pros-y-contras-

a0ec53f786fb

Swan, M. (2015). *Blockchain for a New Economy*. https://doi.org/10.1017/CBO9781107415324.004

Tapscott, Don, Tapscott, Alex, S. A. (2019). *Blockchain Revolution* (Colombo Andina de Impresos (Ed.); 4th ed.). https://www.amazon.es/revolución-blockchain-Descubre-tecnología-transformará-ebook/dp/B01N5TK29G

Thomas, D., & Barry, B. M. (2003). Model driven development: the case for domain oriented programming. *Oopsla*, 2–7. https://doi.org/10.1145/949344.949346

Trejo, J., & Robles, A. (2010). Conceptos fundamentales de Ingeniería dirigida por Modelos y Modelos de Dominio Específico. *Revista de Investigación de Sistemas e Informática*, *7*(2), 9–19.

Triana Casallas, Jenny Alexandra, Rodriguez Molano, José Ignacio, Fuentes, H. J. (2020). MEVF "Fiscal Surveillance Entities model “A DSL proposed for interoperability and management of fiscal scenaries.” *International Journal of Mechanical and Production Engineering Research and Development (IJMPERD)*, *10*(6), 491–498. http://www.tjprc.org/view_paper.php?id=14796

Triana Casallas, J. A., Cueva-Lovelle, J. M., & Rodríguez Molano, J. I. (2020). Smart Contracts with Blockchain in the Public Sector. *International Journal of Interactive Multimedia and Artificial Intelligence*, *6*(3), 63. https://doi.org/10.9781/ijimai.2020.07.005

Tron Fundation. (2018). TRON - Advanced Decentralized Blockchain Platform - Whitepaper Version: 2.0. In *TRON Foundation* (pp. 1–40). https://tron.network/static/doc/white_paper_v_2_0.pdf--

Using Blockchain Technology to Eliminate Corruption in Developing Nations - coinweez. (n.d.). Retrieved May 29, 2020, from https://coinweez.com/using-blockchain-technology-eliminate-corruption-developing-nations/

Vass Company. (2017). *Blockchain, guía rápida de ‘Smart Contracts’ o contratos inteligentes*. https://vasscompany.com/blockchain-guia-rapida-de-smart-contracts-o-contratos-inteligentes/

Vigna, P., & Casey, M. (2015). *The age of cryptocurrency : how bitcoin and digital money are challenging the global economic order*.

Villalobos, J. (2021). *Modelos y Metamodelos Jorge Villalobos , PhD* (Issue February).

Viriyasitavat, W., & Hoonsopon, D. (2019). Blockchain characteristics and consensus in modern business processes. *Journal of Industrial Information*

Integration, *13*, 32–39. https://doi.org/10.1016/j.jii.2018.07.004

What is Blockchain for Business? – IBM Blockchain | IBM. (n.d.). Retrieved May 26, 2021, from https://www.ibm.com/pe-es/blockchain?lnk=hpmps_bubc_eces

ANEXO 1. ANÁLISIS DEL SISTEMA

1. ESPECIFICACIÓN DE REQUERIMIENTOS

1.1. Requerimientos generales del sistema

El sistema debe cumplir con la siguientes características y funcionalidades:

Requerimiento	Descripción	Tipo
Registro de usuarios	Por medio de un formulario registrar los datos de los usuarios que van a interactuar con la plataforma, se crea usuario y contraseña.	Funcional
Acceso de usuarios	A través de la página de login con el usuario y contraseña permitir el acceso de los usuarios a la plataforma.	Funcional
Delimitación de roles	Permite mostrar las funciones acordes a cada usuario, según sus permisos en la realización e interacción con los contratos.	Funcional
Construcción de contratos	Ayuda a la elaboración de los contratos, detallar cada una de las actividades, los participantes, el tiempo y el presupuesto que este requiere.	Funcional
Gestión de presupuesto	Se detalla el destino que tiene el presupuesto en los diferentes contratos y actividades de estos.	Funcional
Edición de contratos	Se puede corregir cualquier punto de un contrato antes de que este sea aceptado.	Funcional
Aceptación de contrato	Una vez todos los detalles del contrato estén en orden, la persona o personas encargadas podrán firmarlo para que empiece su ejecución.	Funcional
Avance de cumplimiento del contrato	Visualización del progreso de un contrato con respecto al tiempo que tiene este para ser culminado.	Funcional
Reporte de acciones realizadas del contrato	Detalle de cada una de las actividades realizadas de un contrato, estado, observaciones y presupuesto gastado.	Funcional
Alertas de incumplimiento	Si existen retrasos en tiempo o gastos fuera de los previstos, se generará una alerta a los encargados de un contrato.	Funcional
Bloqueo de contrato	Si es necesario se puede bloquear un contrato que este siendo incumplido.	Funcional
Resumen de contratos en ejecución	Se puede visualizar los contratos que se estén ejecutando, con un breve resumen de su objetivo principal y el presupuesto que requieren.	Funcional
Generación de contrato nuevo	Usando un contrato ya firmado que deba modificarse, se crea un nuevo contrato con los cambios necesarios.	Funcional
Historial de transacción	Cada una de las actividades realizadas dentro de la plataforma serán registradas en un historial para ayudar en los procesos de auditoria.	Funcional

Investigación de transacciones	Se puede revisar el historial de las transacciones para hacer un seguimiento de un contrato, delimitación de presupuesto o de un usuario.	Funcional
Acceso a la plataforma	El acceso a la plataforma de los contratos inteligentes es por medio de una interfaz web disponible para diferentes dispositivos.	No funcional
Interfaz gráfica de usuarios	La GUI por medio de menú de navegación, ventanas, formularios y botones ayudara a los usuarios a hacer de manera más fácil los procesos.	No funcional
Base de datos	Se hará uso de una base de datos relacional mediante un manejador convencional.	No funcional
Seguridad para los usuarios y contratos	La seguridad de los usuarios y los contratos deberá ser manejada desde la base de datos, haciendo uso de algoritmos de encriptamiento y determinación de roles.	No funcional

1.2. Especificación de requerimientos funcionales

ID: RF-1	*Registro de usuarios*
Versión	*1*
Dependencias	*N/A*
Descripción	Por medio de un formulario registrar los datos de los usuarios que van a interactuar con la plataforma, se crea usuario y contraseña.
Importancia	Alta
Prioridad	Alta

ID: RF-2	*Acceso de usuarios*
Versión	*1*
Dependencias	• *RF-1*
Descripción	A través de la página de login con el usuario y contraseña permitir el acceso de los usuarios a la plataforma.
Importancia	Alta
Prioridad	Alta

ID: RF-3	*Delimitación de roles*
Versión	*1*
Dependencias	*N/A*
Descripción	Permite mostrar las funciones acordes a cada usuario, según sus permisos en la realización e interacción con los contratos.
Importancia	Alta
Prioridad	Media

ID: RF-4	*Construcción de contratos*
Versión	*1*
Dependencias	• *RF-5*
Descripción	Ayuda a la elaboración de los contratos, detallar cada una de las actividades, los participantes, el tiempo y el presupuesto que este requiere.
Importancia	Alta
Prioridad	Alta

ID: RF-5	*Gestión de presupuesto*
Versión	*1*
Dependencias	*N/A*
Descripción	Se detalla el destino que tiene el presupuesto en los diferentes contratos y actividades de estos.
Importancia	Alta
Prioridad	Alta

ID: RF-6	*Edición de contratos*
Versión	*1*
Dependencias	• *RF-4* • *RF-5*
Descripción	Se puede corregir cualquier punto de un contrato antes de que este sea aceptado.
Importancia	Media
Prioridad	Media

ID: RF-7	*Aceptación de contrato*
Versión	*1*
Dependencias	• *RF-4* • *RF-5* • *RF-6*
Descripción	Una vez todos los detalles del contrato estén en orden, la persona o personas encargadas podrán firmarlo para que empiece su ejecución.
Importancia	Alta
Prioridad	Alta

ID: RF-8	*Avance del cumplimiento del contrato*
Versión	*1*
Dependencias	• *RF-7* • *RF-9*
Descripción	Visualización del progreso de un contrato con respecto al tiempo que tiene este para ser culminado.
Importancia	Media
Prioridad	Media

ID: RF-9	*Reporte de actividades realizadas del contrato*
Versión	*1*
Dependencias	• *RF-7*
Descripción	Detalle de cada una de las actividades realizadas de un contrato, estado, observaciones y presupuesto gastado.
Importancia	Media
Prioridad	Media

ID: RF-10	*Alertas de incumplimiento*
Versión	*1*
Dependencias	• *RF-9*
Descripción	Si existen retrasos en tiempo o gastos fuera de los previstos, se generará una alerta a los encargados de un contrato.
Importancia	Alta
Prioridad	Media

ID: RF-11	*Bloqueo de contrato*
Versión	*1*
Dependencias	• *RF-10*
Descripción	Si es necesario se puede bloquear un contrato que este siendo incumplido.
Importancia	Media
Prioridad	Media

ID: RF-12	*Resumen de contratos en ejecución*
Versión	*1*
Dependencias	• *RF-8* • *RF-9*
Descripción	Se puede visualizar los contratos que se estén ejecutando, con un breve resumen de su objetivo principal y el presupuesto que requieren.
Importancia	Media
Prioridad	Media

ID: RF-13	*Generación de contrato nuevo*
Versión	*1*
Dependencias	• *RF-7*
Descripción	Usando un contrato ya firmado que deba modificarse, se crea un nuevo contrato con los cambios necesarios.
Importancia	Alta
Prioridad	Alta

ID: RF-14	*Historial de transacciones*
Versión	*1*
Dependencias	• *RF-1* • *RF-2* • *RF-3* • *RF-4* • *RF-5* • *RF-6* • *RF-7* • *RF-8* • *RF-9* • *RF-10* • *RF-11* • *RF-12* • *RF-13* • *RF-15*
Descripción	Cada una de las actividades realizadas dentro de la plataforma serán registradas en un historial para ayudar en los procesos de auditoria.
Importancia	Alta
Prioridad	Alta

ID: RF-15	*Investigación de transacciones*
Versión	*1*
Dependencias	• *RF-14*
Descripción	Se puede revisar el historial de las transacciones para hacer un seguimiento de un contrato, delimitación de presupuesto o de un usuario.
Importancia	Alta
Prioridad	Alta

1.3. Especificación de requerimientos no funcionales

ID: RNF-1	*Acceso a la plataforma*
Versión	*1*
Descripción	El acceso a la plataforma de los contratos inteligentes es por medio de una interfaz web disponible para diferentes dispositivos.
Importancia	Alta
Prioridad	Media

ID: RNF-2	*Interfaz gráfica de usuarios*
Versión	*1*
Descripción	La GUI por medio de menú de navegación, ventanas, formularios y botones ayudara a los usuarios a hacer de manera más fácil los procesos.
Importancia	Alta
Prioridad	Media

ID: RNF-3	*Base de datos*
Versión	*1*
Descripción	Se hará uso de una base de datos relacional mediante un manejador convencional.
Importancia	Alta
Prioridad	Media

ID: RNF-4	*Seguridad para los usuarios y los contratos*
Versión	*1*
Descripción	La seguridad de usuarios y contratos se maneja desde la base de datos, haciendo uso de algoritmos de encriptamiento y determinación de roles.
Importancia	Alta
Prioridad	Media

1.4. Especificación de actores

- *Contratante*

Persona encargada de elaborar los contratos, destinación de presupuesto y delimitación de actividades. Puede visualizar el avance de un contrato y de ser necesario bloquearlo ante un incumplimiento.

- *Contratista*

Usuario que puede visualizar el contrato, firmar el contrato, reportar actividades realizadas o inconvenientes que pueda tener.

- *Auditor*

Encargado de revisar porque un contrato fue bloqueado y debe decidir qué acciones se deben seguir.

2. IDENTIFICACIÓN DE CASOS DE USO

Código	Nombre	Casos de uso relacionados	Actores
CU-1	Registro de usuarios	-	• Contratante • Contratista • Auditor
CU-2	Acceso de usuarios	• CU-1	• Contratante • Contratista • Auditor
CU-3	Delimitación de roles	-	-
CU-4	Construcción de contratos	• CU-5	• Contratante
CU-5	Gestión de presupuesto	-	• Contratante
CU-6	Edición de contratos	• CU-4 • CU-5	• Contratante
CU-7	Aceptación de contrato	• CU-4 • CU-5 • CU-6	• Contratante • Contratista
CU-8	Avance de cumplimiento del contrato	• CU-7 • CU-9	• Contratante
CU-9	Reporte de acciones realizadas del contrato	• CU-7	• Contratista
CU-10	Alertas de incumplimiento	• CU-9	-
CU-11	Bloqueo de contrato	• CU-10	• Contratante
CU-12	Resumen de contratos en ejecución	• CU-8 • CU-9	-
CU-13	Generación de contrato nuevo	• CU-7	• Contratante
CU-14	Historial de transacciones	• CU-1 • CU-2 • CU-3 • CU-4 • CU-5 • CU-6 • CU-7 • CU-8 • CU-9 • CU-10 • CU-11 • CU-12 • CU-13 • CU-15	
CU-15	Investigación de transacciones	• CU-14	• Auditor

2.1. Priorización de casos de uso

Orden de prioridad	Código	Nombre	Cantidad de dependencias
1	CU-14	Historial de transacciones	14
2	CU-7	Aceptación de contrato	3
3	CU-6	Edición de contratos	2
4	CU-8	Avance del cumplimiento del contrato	2
5	CU-12	Resumen de contratos de ejecución	2
6	CU-2	Acceso de usuarios	1
7	CU-4	Construcción de contratos	1
8	CU-9	Reporte de acciones realizadas del contrato	1
9	CU-10	Alertas de incumplimiento	1
10	CU-11	Bloqueo de contrato	1
11	CU-13	Generación de contrato nuevo	1
12	CU-15	Investigación de transacciones	1
13	CU-1	Registro de usuarios	0
14	CU-3	Delimitación de roles	0
15	CU-5	Gestión de presupuesto	0

2.2. Especificación de casos de uso

ID	**CU-1**	
Nombre	Registro de usuarios	
Actores ➢ Contratante ➢ Contratista ➢ Auditor		
Resumen Por medio de un formulario registrar los datos de los usuarios que van a interactuar con la plataforma, se crea usuario y contraseña.		
Precondiciones		
Actor	**Software**	
1. Ingresa datos personales, correo y contraseña. 3. Registra los datos en la plataforma.	2. Valida la seguridad de la contraseña. 4. Guarda los datos en la base de datos para permitir el acceso del usuario a la plataforma.	
Postcondiciones ➢ Se cuenta con u nuevo usuario en la plataforma.		

ID	**CU-2**
Nombre	Acceso de usuarios
Actores ➢ Contratante ➢ Contratista ➢ Auditor	

Resumen A través de la página de login con el usuario y contraseña permitir el acceso de los usuarios a la plataforma.	
Precondiciones ➢ Los usuarios deben estar registrados en la plataforma	
Actor	**Software**
1. Ingresar al login de la plataforma e ingresa usuario, contraseña y valida sus credenciales. 3. Accede a la plataforma o corrige datos de acceso.	2. Valida los datos ingresados: ➢ Si el usuario esta registrado. ➢ Si la contraseña es válida al usuario. ➢ Permite o denegar el acceso.
Postcondiciones ➢ Puede interactuar con los módulos de la plataforma según el rol que el usuario tenga.	

ID	**CU-3**
Nombre	Delimitación de roles
Actores	
Resumen Permite mostrar las funciones acordes a cada usuario, según sus permisos en la realización e interacción con los contratos.	
Precondiciones	
Actor	**Software**
	1. Muestra los módulos con los que puede interactuar cada rol.
Postcondiciones ➢ Cada usuario cumple una función especifica dentro de la plataforma.	

ID	**CU-4**
Nombre	Construcción de contratos
Actores ➢ Contratante	
Resumen Ayuda a la elaboración de los contratos, detallar cada una de las actividades, los participantes, el tiempo y el presupuesto que este requiere.	
Precondiciones ➢ Tener definido la gestión del presupuesto	
Actor	**Software**
1. Entrar al módulo de construcción de contrato. 3. Se llenan los datos del formulario y se pasa al detalle del contrato. 5. Se ingresan los datos de las actividades y se acepta.	2. Mostrar el formulario inicial donde se elige que parte del presupuesto será involucrado, nombre del contrato, participantes y periodo de ejecución. 4. Se genera el formulario de ingreso de actividades del contrato, con nombre, periodo de ejecución y presupuesto. 6. Se guarda la información en base de datos y se muestra una visualización del contrato.
Postcondiciones ➢ Se crea el contrato y queda listo para firmar.	

ID	**CU-5**
Nombre	Gestión de presupuesto
Actores ➢ Contratante	
Resumen Se detalla el destino que tiene el presupuesto en los diferentes contratos y actividades de estos.	
Precondiciones	
Actor	**Software**
1. Ingresa al módulo de gestión de presupuestos. 3. Registra los datos y se dirige al siguiente formulario. 4. Escoge el destino del presupuesto y acepta.	2. Muestra formulario donde se requieren los datos de periodo y suma de dinero. 4. Muestra el formulario donde se particiona y selecciona el destino del presupuesto.
Postcondiciones ➢ Se puede hacer la creación contratos con presupuestos ya definidos.	

ID	**CU-6**
Nombre	Edición de contratos
Actores ➢ Contratante	
Resumen Se puede corregir cualquier punto de un contrato antes de que este sea aceptado.	
Precondiciones ➢ El contrato debe estar construido, pero no firmado. ➢ Se debe haber gestionado el presupuesto.	
Actor	**Software**
1. Selecciona el contrato el cual va a editar. 3. Edita los datos generales y pasa a las actividades. 5. Modifica las actividades que necesite y acepta.	2. Se muestra el formulario inicial con los datos originales que se editaran. 4. Muestra las actividades con los datos originales que se editaran. 6. Actualiza los datos y muestra una visualización del contrato.
Postcondiciones ➢ Se crea el contrato y queda listo para firmar.	

ID	**CU-7**
Nombre	Aceptación de contratos
Actores ➢ Contratante ➢ Contratista	
Resumen Una vez todos los detalles del contrato estén en orden, la persona o personas encargadas podrán firmarlo para que empiece su ejecución.	
Precondiciones ➢ Se debe contar con el presupuesto ya gestionado. ➢ El contrato debe estar construido en su versión final.	
Actor	**Software**

1. El usuario debe ingresar a visualizar los contratos ya realizados. 3. Seleccionar el contrato a firmar. 5. Firmar y aceptar el contrato.	2. Mostrar el resumen de los contratos ya construidos donde participa el usuario. 4. Permitir la firma del usuario. 6. Validar si el contrato fue firmado por todos los participantes e iniciar la ejecución del contrato.
Postcondiciones ➢ Se inicia la ejecución de un contrato.	

ID	**CU-8**
Nombre	Avance de cumplimiento de contrato
Actores ➢ Contratante	
Resumen Visualización del progreso de un contrato con respecto al tiempo que tiene este para ser culminado.	
Precondiciones ➢ El contrato debe estar firmado. ➢ Se debe reportar las actividades realizadas de cada contrato.	
Actor	**Software**
1. El usuario ingresa a ejecución de contratos. 3. Selecciona el contrato que desea visualizar.	2. Se muestran los contratos donde el usuario participa. 4. Se muestra el detalle del contrato, actividades realizadas, periodo de ejecución y presupuesto gastado.
Postcondiciones ➢ Se lleva un control sobre el contrato.	

ID	**CU-9**
Nombre	Reporte actividades realizadas del contrato
Actores ➢ Contratista	
Resumen Detalle de cada una de las actividades realizadas de un contrato, estado, observaciones y presupuesto gastado.	
Precondiciones ➢ El contrato debe estar firmado.	
Actor	**Software**
1. El usuario ingresa a reportes de actividades. 3. Llena el formulario para que se reporten las actividades realizadas.	2. Se muestra un formulario con las actividades donde se ingresan las observaciones que se tienen y si fue cumplida. 4. Se actualiza el estado de ejecución del contrato.
Postcondiciones ➢ Se visualiza el estado de las actividades de cada contrato.	

ID	**CU-10**
Nombre	Alertas de incumplimiento
Actores	
Resumen Si existen retrasos en tiempo o gastos fuera de los previstos, se generará una alerta a los encargados de un contrato.	

Precondiciones ➢ Se debe generar reportes de las actividades.	
Actor	**Software**
	1. Al no tener reportes de actividades en las fechas establecidas se generan alertas al contratante para que tome las respectivas medidas.
Postcondiciones ➢ El contratante puede tomar medidas en la ejecución de un contrato.	

ID	**CU-11**
Nombre	Bloqueo de contrato
Actores ➢ Contratante	
Resumen Si es necesario se puede bloquear un contrato que este siendo incumplido.	
Precondiciones ➢ Deben existir alertas de incumplimiento.	
Actor	**Software**
1. Ingresar a contratos en ejecución. 3. Bloquear contrato.	2. Mostrar el resumen de los contratos con la opción de bloquear contratos. 3. Quitar el contrato de los registros de ejecución.
Postcondiciones ➢ El contrato entra en proceso de investigación.	

ID	**CU-12**
Nombre	Resumen de contratos
Actores	
Resumen Se puede visualizar los contratos que se estén ejecutando, con un breve resumen de su objetivo principal y el presupuesto que requieren.	
Precondiciones ➢ Se debe contar con el reporte de las actividades realizadas. ➢ El avance del contrato.	
Actor	**Software**
	1. Mostrar el resumen y detalle de cada uno de los contratos en ejecución.
Postcondiciones ➢ Se controla cada uno de los contratos.	

ID	**CU-13**
Nombre	Gestión de contrato nuevo.
Actores ➢ Contratante	
Resumen Usando un contrato ya firmado que deba modificarse, se crea un nuevo contrato con los cambios necesarios.	
Precondiciones ➢ Se debe contar con un contrato firmado.	
Actor	**Software**

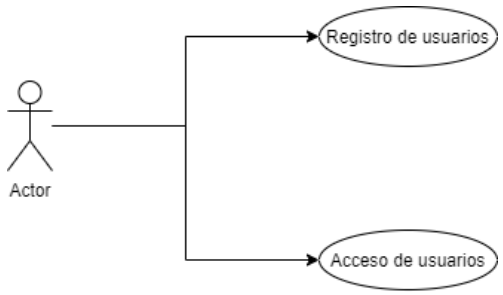
1. Entra a generación de nuevo contrato. 3. Selecciona el contrato base a con el que se generara el nuevo contrato. 5. Se llenan los formularios y se acepta.	2. Muestra los contratos firmados. 4. Se tren los respectivos formularios. 6. Se guarda la información del nuevo contrato y se pasa a contratos por firmar.
Postcondiciones ➢ Se tiene el contrato listo para la firma.	

ID	**CU-14**
Nombre	Historial de transacciones
Actores	
Resumen Cada una de las actividades realizadas dentro de la plataforma serán registradas en un historial para ayudar en los procesos de auditoria.	
Precondiciones ➢ La actividad realizada en todos los demás casos de usos	
Actor	**Software**
	1. Se registra cada transacción realizada en la base de datos con su respectivo hash.
Postcondiciones ➢ Se tiene el historial de todo lo realizado dentro de la plataforma.	

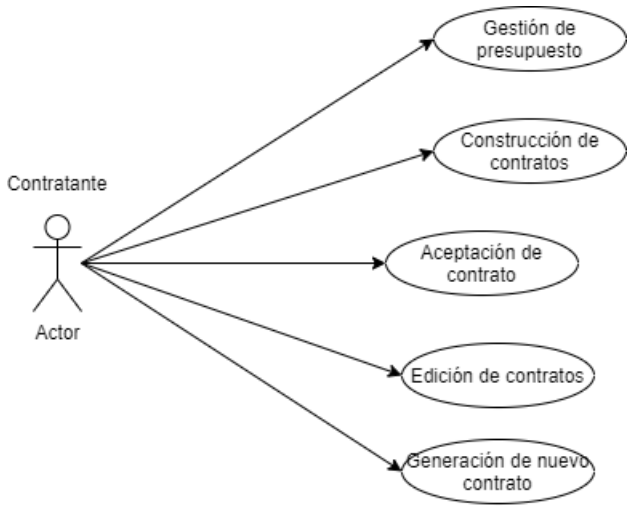
ID	**CU-15**
Nombre	Investigación de transacciones.
Actores ➢ Auditor	
Resumen Se puede revisar el historial de las transacciones para hacer un seguimiento de un contrato, delimitación de presupuesto o de un usuario.	
Precondiciones ➢ Se debe contar con el historial de transacciones.	
Actor	**Software**
1. Entra a la revisión de transacciones. 3. Selecciona los parámetros de su búsqueda. 5. Revisa los datos y requiere PDF con la información.	2. Muestra los filtros de búsqueda. 4. Trae todos los datos que necesita el usuario. 6. Genera el documento que necesita el usuario.
Postcondiciones ➢ Se tiene una auditoria de los procesos.	

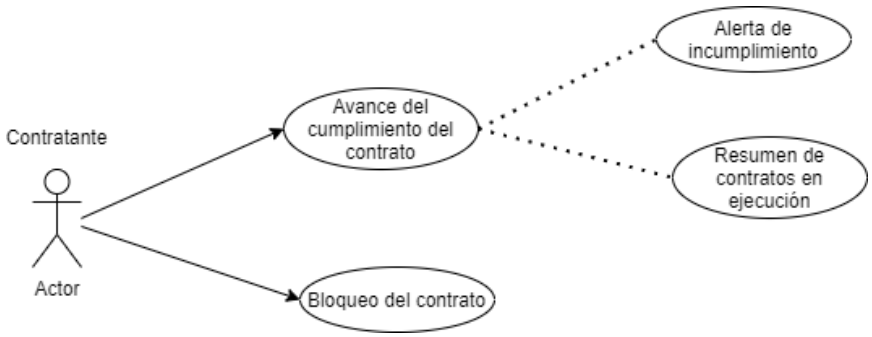
2.3. Diagramas de caso de uso

- Acceso a la plataforma



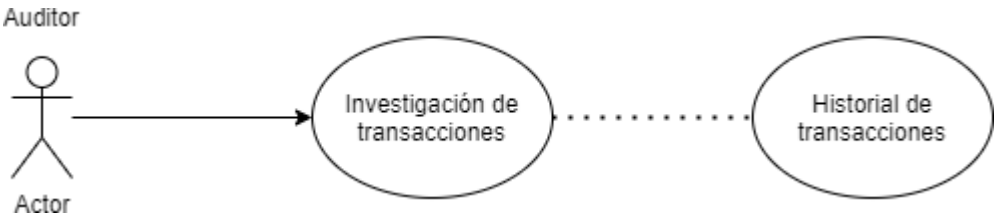
- Acciones del contratante





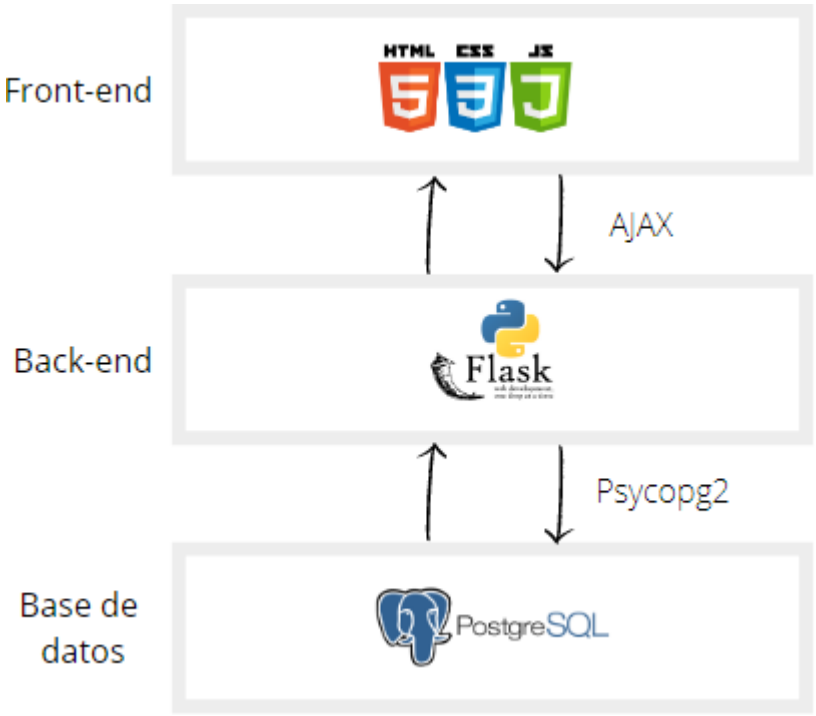
- Acciones del contratista

- Acciones del auditor

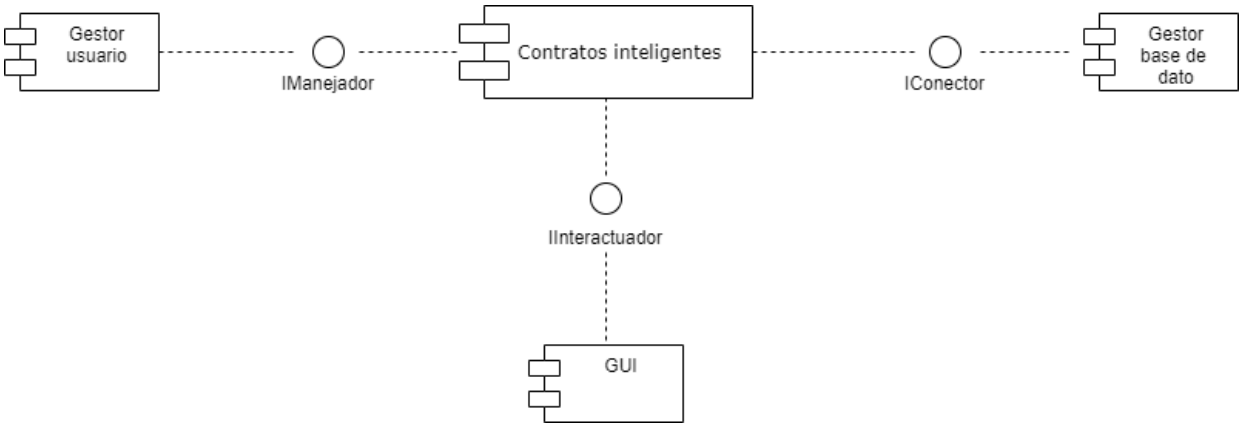


3. DISEÑO ESTRUCTURAL

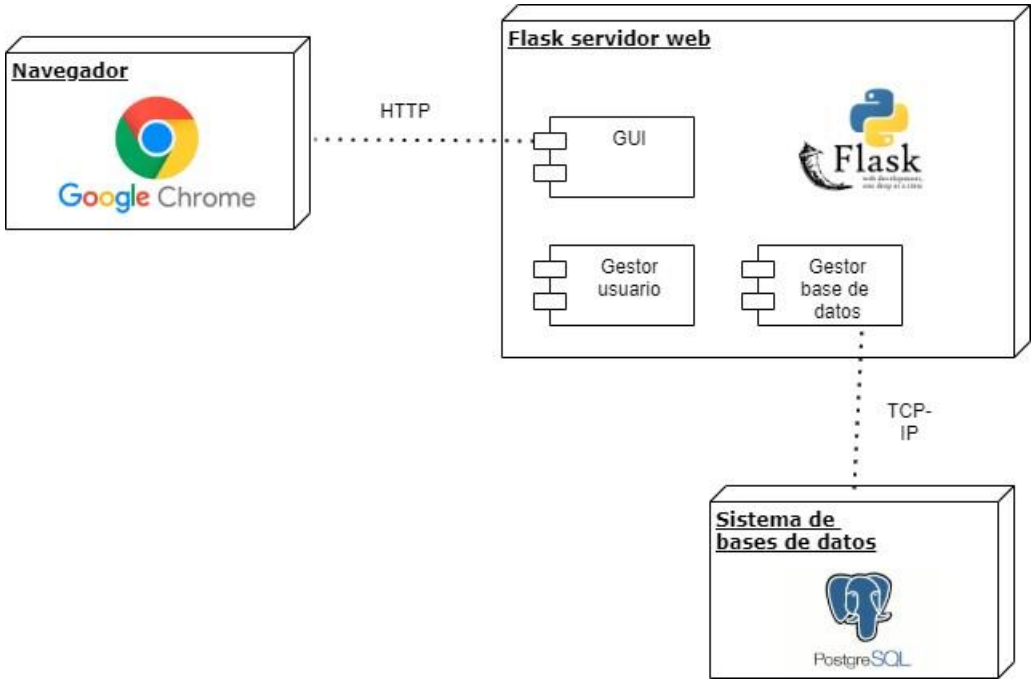
3.1. Arquitectura



3.2. Diagrama de componentes



3.3. Diagrama de despliegue



ANEXO 2. REPORTE DE AUDITORÍA

Reporte
No.98758766878a904de0fe5cd591e2260015d9b70adf99441be3d84cb0bf7b47c9
Generado - Fecha: 2021-06-17 Hora: 02:44:21
Por: 52919099 - Auditor Ciudadano

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623897734.3303068
Usuario: CMV
ID (contrato/presupuesto):
a956e32a778ae7cc521864827d8bdc8835443ebca02acf630b982dd76ad7ffb0Nombre: Servicios profesionales de apoyo jurídico mes 1
Descripción: Apoyar en el seguimiento, control y mejoramiento de los procesos jurídicos, controversias judicialesEstado: S
Presupuesto: 2400000
Ejecutado: 2400000
Disponible: 0
Descripción de transacción: Registro actividad
Fecha de inicio: 2021-05-06
Fecha de fin: 2021-06-05

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623897734.2483304
Usuario: CMV
ID (contrato/presupuesto):
a956e32a778ae7cc521864827d8bdc8835443ebca02acf630b982dd76ad7ffb0Nombre: Servicios profesionales de asesoría jurídica
Descripción: PRESTACION DE SERVICIOS DE APOYO A LA GESTION PARA REALIZAR ACTIVIDADE
Estado: Activo
Presupuesto: 9600000
Ejecutado: 2400000
Disponible: 7200000
Descripción de transacción: Movimiento de contratoFecha de inicio: 2021-05-06
Fecha de fin: 2021-09-05

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623897734.1707397
Usuario: CMV
ID (contrato/presupuesto):
786c06452cc4ef3a9a5d3275b28c1eb9e34bdcfe717938ca9202a5818e49b12eNombre: Honorarios
Descripción: 2021
Estado: Activo
Presupuesto: 42535000
Ejecutado: 2400000
Disponible: 40135000
Descripción de transacción: Movimiento
presupuestoFecha de inicio: 2021-01-01
Fecha de fin: 2021-12-31

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623897684.8554044
Usuario: espejo
ID (contrato/presupuesto):
a956e32a778ae7cc521864827d8bdc8835443ebca02acf630b982dd76ad7ffb0Nombre: Servicios profesionales de asesoría jurídica
Descripción: PRESTACION DE SERVICIOS DE APOYO A LA GESTION PARA REALIZAR ACTIVIDADE
Estado: Activo
Presupuesto: 9600000
Ejecutado: 2400000
Disponible: 7200000
Descripción de transacción: Actividad parcial
Fecha de inicio: 2021-05-06
Fecha de fin: 2021-09-05

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623897406.127663
Usuario: CMV
ID (contrato/presupuesto):
a956e32a778ae7cc521864827d8bdc8835443ebca02acf630b982dd76ad7ffb0Nombre: Servicios profesionales de apoyo jurídico mes 3
Descripción: Apoyar en el seguimiento, control y mejoramiento de los procesos jurídicos, controversias judicialesEstado: N
Presupuesto: 2400000
Ejecutado: 0
Disponible: 2400000
Descripción de transacción: Creación actividad
Fecha de inicio: 2021-07-06
Fecha de fin: 2021-08-05

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623897406.125402
Usuario: CMV
ID (contrato/presupuesto):
a956e32a778ae7cc521864827d8bdc8835443ebca02acf630b982dd76ad7ffb0Nombre: Servicios profesionales de apoyo jurídico mes 1
Descripción: Apoyar en el seguimiento, control y mejoramiento de los procesos jurídicos, controversias judicialesEstado: N
Presupuesto: 2400000
Ejecutado: 0
Disponible: 2400000
Descripción de transacción: Creación actividad
Fecha de inicio: 2021-05-06
Fecha de fin: 2021-06-05

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623897406.1161408
Usuario: CMV
ID (contrato/presupuesto):
a956e32a778ae7cc521864827d8bdc8835443ebca02acf630b982dd76ad7ffb0Nombre: Servicios profesionales de apoyo jurídico mes 4
Descripción: Apoyar en el seguimiento, control y mejoramiento de los procesos jurídicos, controversias judicialesEstado: N
Presupuesto: 2400000
Ejecutado: 0
Disponible: 2400000
Descripción de transacción: Creación actividad
Fecha de inicio: 2021-08-06
Fecha de fin: 2021-09-05

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623897406.0958397
Usuario: CMV
ID (contrato/presupuesto):
a956e32a778ae7cc521864827d8bdc8835443ebca02acf630b982dd76ad7ffb0Nombre: Servicios profesionales de apoyo jurídico mes 2
Descripción: Apoyar en el seguimiento, control y mejoramiento de los procesos jurídicos, controversias judicialesEstado: N
Presupuesto: 2400000
Ejecutado: 0
Disponible: 2400000
Descripción de transacción: Creación actividad
Fecha de inicio: 2021-06-06
Fecha de fin: 2021-07-05

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623896959.732064
Usuario: CMV
ID (contrato/presupuesto):
a956e32a778ae7cc521864827d8bdc8835443ebca02acf630b982dd76ad7ffb0Nombre: Servicios profesionales de asesoría jurídica
Descripción: PRESTACION DE SERVICIOS DE APOYO A LA GESTION PARA REALIZAR ACTIVIDADE
Estado: Sin firmar
Presupuesto: 9600000
Ejecutado: 0
Disponible: 9600000
Descripción de transacción: Realización de contratoFecha de inicio: 2021-05-06
Fecha de fin: 2021-09-05

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623896723.902828
Usuario: CMV
ID (contrato/presupuesto):
4c1a468412e95bf5eafc308c70688bcad74e89a92054ed0bd8d592a3ce78a0ecNombre: Gestión financiera y contable mes 2
Descripción: Gestión financiera y contable abril-mayo
Estado: S
Presupuesto: 3600000
Ejecutado: 3600000
Disponible: 0
Descripción de transacción: Registro actividad
Fecha de inicio: 2021-04-16
Fecha de fin: 2021-05-15

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623896723.8202875
Usuario: CMV
ID (contrato/presupuesto):
4c1a468412e95bf5eafc308c70688bcad74e89a92054ed0bd8d592a3ce78a0ecNombre: Servicios profesionales especializados contador
Descripción: CONTRATO DE PRESTACION DE SERVICIOS PROFESIONALES DE UN CONTADOR PUB
Estado: Activo
Presupuesto: 14880000
Ejecutado: 3600000
Disponible: 11280000
Descripción de transacción: Movimiento de contratoFecha de inicio: 2021-03-16
Fecha de fin: 2021-07-20

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623896723.7383888
Usuario: CMV
ID (contrato/presupuesto):
786c06452cc4ef3a9a5d3275b28c1eb9e34bdcfe717938ca9202a5818e49b12eNombre: Honorarios
Descripción: 2021
Estado: Activo
Presupuesto: 42535000
Ejecutado: 3600000
Disponible: 38935000
Descripcion de transaccion: Movimiento
presupuestoFecha de inicio: 2021-01-01
Fecha de fin: 2021-12-31

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623896600.6698208
Usuario: oriana
ID (contrato/presupuesto):
4c1a468412e95bf5eafc308c70688bcad74e89a92054ed0bd8d592a3ce78a0ecNombre: Servicios profesionales especializados contador
Descripción: CONTRATO DE PRESTACION DE SERVICIOS PROFESIONALES DE UN CONTADOR PUB
Estado: Activo
Presupuesto: 14520000
Ejecutado: 3600000
Disponible: 10920000
Descripción de transacción: Actividad parcial
Fecha de inicio: 2021-03-16
Fecha de fin: 2021-07-20

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623896489.3959928
Usuario: oriana
ID (contrato/presupuesto):
4c1a468412e95bf5eafc308c70688bcad74e89a92054ed0bd8d592a3ce78a0ecNombre: Servicios profesionales especializados contador
Descripción: CONTRATO DE PRESTACION DE SERVICIOS PROFESIONALES DE UN CONTADOR PUB
Estado: Activo
Presupuesto: 14880000
Ejecutado: 360000
Disponible: 14520000
Descripción de transacción: Actividad parcial
Fecha de inicio: 2021-03-16
Fecha de fin: 2021-07-20

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623896170.7330298
Usuario: CMV
ID (contrato/presupuesto):
4c1a468412e95bf5eafc308c70688bcad74e89a92054ed0bd8d592a3ce78a0ecNombre: Gestión financiera y contable mes 3
Descripción: Gestión financiera y contable mayo-junio
Estado: N
Presupuesto: 3600000
Ejecutado: 0
Disponible: 3600000
Descripción de transacción: Creación actividad
Fecha de inicio: 2021-05-16
Fecha de fin: 2021-06-15

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623896170.7307045
Usuario: CMV
ID (contrato/presupuesto):
4c1a468412e95bf5eafc308c70688bcad74e89a92054ed0bd8d592a3ce78a0ecNombre: Gestión financiera y contable mes 1
Descripción: Gestión financiera y contable marzo 16 a 15 de abril
Estado: N
Presupuesto: 3600000
Ejecutado: 0
Disponible: 3600000
Descripción de transacción: Creación actividad
Fecha de inicio: 2021-03-16
Fecha de fin: 2021-04-15

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623896170.7274022
Usuario: CMV
ID (contrato/presupuesto):
4c1a468412e95bf5eafc308c70688bcad74e89a92054ed0bd8d592a3ce78a0ecNombre: Gestión financiera y contable días
Descripción: Gestión financiera y contable julio
Estado: N
Presupuesto: 480000
Ejecutado: 0
Disponible: 480000
Descripción de transacción: Creación actividad
Fecha de inicio: 2021-07-16
Fecha de fin: 2021-07-20

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623896170.7144523
Usuario: CMV
ID (contrato/presupuesto):
4c1a468412e95bf5eafc308c70688bcad74e89a92054ed0bd8d592a3ce78a0ecNombre: Gestión financiera y contable mes 2
Descripción: Gestión financiera y contable abril-mayo
Estado: N
Presupuesto: 3600000
Ejecutado: 0
Disponible: 3600000
Descripción de transacción: Creación actividad
Fecha de inicio: 2021-04-16
Fecha de fin: 2021-05-15

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623896170.700575
Usuario: CMV
ID (contrato/presupuesto):
4c1a468412e95bf5eafc308c70688bcad74e89a92054ed0bd8d592a3ce78a0ecNombre: Gestión financiera y contable mes 4
Descripción: Gestión financiera y contable junio-julio
Estado: N
Presupuesto: 3600000
Ejecutado: 0
Disponible: 3600000
Descripción de transacción: Creación actividad
Fecha de inicio: 2021-06-16
Fecha de fin: 2021-07-15

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623895788.5173037
Usuario: CMV
ID (contrato/presupuesto):
4c1a468412e95bf5eafc308c70688bcad74e89a92054ed0bd8d592a3ce78a0ecNombre: Servicios profesionales especializados contador
Descripción: CONTRATO DE PRESTACION DE SERVICIOS PROFESIONALES DE UN CONTADOR PUB
Estado: Sin firmar
Presupuesto: 14880000
Ejecutado: 0
Disponible: 14880000
Descripción de transacción: Realización de
contratoFecha de inicio: 2021-03-16
Fecha de fin: 2021-07-20

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623895172.7090917
Usuario: CMV
ID (contrato/presupuesto):
93d28d7b767cb2a31f9c89f69756c4c83193232a80decdae769119f3ed18bca1Nombre: Servicios profesionales especializados contador
Descripción: CONTRATO DE PRESTACION DE SERVICIOS PROFESIONALES DE UN CONTADOR PUB
Estado: Sin firmar
Presupuesto: 14880000
Ejecutado: 0
Disponible: 14880000
Descripción de transacción: Realización de
contratoFecha de inicio: 2021-03-16
Fecha de fin: 2021-07-20

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623894454.5538108
Usuario: CMV
ID (contrato/presupuesto):
ae40cd673d38ee3c1f635707e2106b7a18cc8eb76e8a13b7e5ef52757611a2cbNombre: Adquisición de pólizas para la CMV o sus bienes
Descripción: Adquisición de soat, póliza de seguros para vehículos de la CMV, póliza bienes muebles e inmueble
Estado: N
Presupuesto: 8561532
Ejecutado: 0
Disponible: 8561532
Descripción de transacción: Creación actividad
Fecha de inicio: 2021-02-11
Fecha de fin: 2021-12-31

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623894143.3970652
Usuario: CMV
ID (contrato/presupuesto):
ae40cd673d38ee3c1f635707e2106b7a18cc8eb76e8a13b7e5ef52757611a2cbNombre: Seguros
Descripción: ADQUISICION UNA POLIZA DE SEGUROS PARA LA ENTIDAD, POLIZA BIENES MUEBL
Estado: Sin firmar
Presupuesto: 8561532
Ejecutado: 0
Disponible: 8561532
Descripción de transacción: Realización de contratoFecha de inicio: 2021-02-11
Fecha de fin: 2021-12-31

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623893740.617778
Usuario: CMV
ID (contrato/presupuesto):
d49bda806a7569223584a8ded1ce2d3781c57bafcf3e2fbab638ff1317772a55Nombre: Entregar correspondencia
Descripción: Entregar correspondencia en los destinos que se requiera sean estos urbanos o rurales a nivel nacionEstado: N
Presupuesto: 600000
Ejecutado: 0
Disponible: 600000
Descripción de transacción: Creación actividad
Fecha de inicio: 2021-05-05
Fecha de fin: 2021-12-31

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623893631.3263667
Usuario: CMV
ID (contrato/presupuesto):
d49bda806a7569223584a8ded1ce2d3781c57bafcf3e2fbab638ff1317772a55Nombre: Correspondencia
Descripción: SERVICIO DE RECOLECCION, CURSO Y ENTREGA DE CORRESPONDENCIA Y DEMAS
Estado: Sin firmar
Presupuesto: 600000
Ejecutado: 0
Disponible: 600000
Descripción de transacción: Realización de contratoFecha de inicio: 2021-05-05
Fecha de fin: 2021-12-31

Hash:
000c8cf904627e2547e92f3743cdde43eb43322f1d05d2012d1a86e588203126
Bloque: 1
Timestamp: 1623893546.5163295
Usuario: CMV
ID (contrato/presupuesto):
48c48d493c3eee269e2423dd765638ed7167555de6444ffadc4718bb2f01b7cdNombre:
Correspondencia
Descripción: SERVICIO DE RECOLECCION, CURSO Y ENTREGA DE CORRESPONDENCIA Y DEMAS
Estado: Sin firmar
Presupuesto: 600000
Ejecutado: 0
Disponible: 600000
Descripción de transacción: Realización de
contratoFecha de inicio: 2021-05-05
Fecha de fin: 2021-12-31