



**COMILLAS**  
UNIVERSIDAD PONTIFICIA

ICAI

# GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

TRABAJO FIN DE GRADO

## APLICACIONES DE LOS SMART CONTRACTS A LAS LICENCIAS DE PATENTES

Autor: Juan Monge García Del Pozo

Director: Antonio García de Garmendia

Madrid

Mayo 2023

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título  
*“Aplicaciones de los Smart Contracts a las Licencias de Patentes”*  
en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el  
curso académico 2022/2023 es de mi autoría, original e inédito y  
no ha sido presentado con anterioridad a otros efectos.

El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido  
tomada de otros documentos está debidamente referenciada.

Fdo.: Juan Monge García del Pozo

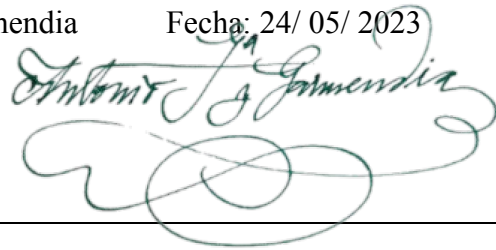
Fecha: 24/ 05/ 2023

Autorizada la entrega del proyecto

EL DIRECTOR DEL PROYECTO

Fdo.: Antonio García de Garmendia

Fecha: 24/ 05/ 2023

A handwritten signature in blue ink, reading "Antonio García de Garmendia", with a large, decorative flourish underneath.

# **APLICACIONES DE LOS SMART CONTRACTS A LAS LICENCIAS DE PATENTES**

**Autor: Monge García del Pozo, Juan**

Director: García de Garmendia, Antonio

Entidad Colaboradora: ICAI- Universidad Pontificia Comillas

## **Resumen del Proyecto:**

Motivado por la idea de combinar las nuevas tecnologías, en este caso la tecnología blockchain con el Sistema de patentes tradicional y motivado también por la existente necesidad de automatizar y optimizar los procesos relacionados con las patentes, como se ha desarrollado en el capítulo 2, se ha llevado a cabo el desarrollo del Presente Proyecto de Fin de Carrera.

Se han dedicado los dos primeros capítulos del proyecto a presentar la situación actual que refiere a la tecnología blockchain y al sistema de patentes. En el primer capítulo se ha llevado a cabo una introducción teórica de los conceptos que después, se vendrán utilizando de forma recurrente a lo largo del proyecto. También en el segundo capítulo se han analizado distintas aplicaciones que existen en la actualidad de la tecnología blockchain y cuyas características, en algunos casos, nos servirán en el siguiente capítulo como bases para desarrollar el Modelo. Por otro lado, en el segundo capítulo, se han estudiado distintos casos de acuerdos de licencias cruzadas para después obtener conclusiones sobre ellos.

El Modelo desarrollado en este Proyecto durante el capítulo 3, cuenta su desarrollo teórico y matemático, ambos se llevan a cabo de forma simultánea para la simplificar así la comprensión de este. El Modelo cuenta con tres funciones principales: Solicitar la explotación de una patente, Gestionar un acuerdo de licencias cruzadas y Crear una nueva patente. Cada una de estas funciones se ha desarrollado de manera similar. En primer lugar, se ha elaborado un diagrama de flujos, donde se esquematiza el funcionamiento de cada función, seguido de este esquema se han descrito y explicado las variables de funcionamiento de cada una de las tres funciones, especificando y detallando la información contenida en cada variable.

Por otro lado, en el Capítulo 3 también, se ha desarrollado una calculadora de precios para las patentes, cuya principal funcionalidad es, de forma objetiva y automática, en

base a unos parámetros que se explican y desarrollan en el capítulo, calcular el precio de cada licencia de una patente.

En el capítulo 4 se ha puesto en práctica el modelo para cada una de las distintas funciones, adaptando los datos reales a nuestro modelo, para asegurar el correcto funcionamiento del mismo. También se han diseñado e incluido capturas de la interfaz que vería el usuario y la OEPM al interactuar con el Modelo desarrollado. Entre las partes desarrolladas de cada caso, se encuentra también el cálculo de los precios de las patentes elaborado por el algoritmo detrás de la calculadora desarrollada en el capítulo anterior.

Una vez planteadas las aplicaciones prácticas del modelo, en el capítulo 5 se ha llevado a cabo una memoria económica donde se estudia el coste de la implementación del Modelo a casos reales. Los costes principales serían de desarrollo auditoria y mantenimiento. El Modelo se ha elaborado con intención de escalar en un futuro como se ha detallado en este capítulo.

Por último, en el último capítulo, se han incluido unas conclusiones del Proyecto, donde se han recopilado las principales ideas desarrolladas a lo largo de los distintos capítulos. Por otro lado, se han incluido también recomendaciones futuras para el Modelo, con el fin de asegurar su viabilidad el futuro e incorporar el uso de nuevas tecnologías.

## **Project Summary:**

Motivated by the idea of combining new technologies, in this case blockchain technology with the traditional patent system and motivated by the existing need to automate and optimize the processes related to patents, as developed in chapter 2, I have carried out the development of this Final Degree Project.

The first two chapters of the project have been dedicated to presenting the current situation regarding blockchain technology and the patent system. In the first chapter, a theoretical introduction of the concepts that will be used recurrently throughout the project has been conducted. Also, in the second chapter we have analyzed different applications that currently exist of blockchain technology and whose characteristics, in some cases, will serve in the next chapter as the basis for developing the Model. On the other hand, in the second chapter, different cases of cross-licensing agreements have been studied in order to draw conclusions about them.

The Model developed in this Project during chapter 3, has its theoretical and mathematical development, both are developed simultaneously in order to simplify its understanding. The Model has three main functions: Requesting the exploitation of a patent, Managing a cross-licensing agreement and Creating a new patent. Each of these functions has been developed in a similar way. First, a flow diagram has been elaborated, where the operation of each function is schematized, followed by a description and explanation of the operating variables of each of the three functions, specifying and detailing the information contained in each variable.

On the other hand, in Chapter 3, a price calculator for patents has also been developed, whose main function is to calculate the price of each patent license in an objective and automatic way, based on some parameters that are explained and developed in the chapter.

In chapter 4, the model has been implemented for each of the different functions, adapting the real data to our model, to ensure its correct operation. We have also designed and included screenshots of the interface that the user and the OEPM would see when interacting with the developed Model. Among the developed parts of each case, there is also the calculation of the patent prices elaborated by the algorithm behind the calculator developed in the previous chapter.

Once the practical applications of the model have been proposed, in chapter 5 an economic report has been carried out where the cost of the implementation of the Model to real cases is studied. The main costs would be development, auditing and maintenance. The Model has been developed with the intention of scaling up in the future as detailed in this chapter.

Finally, in the last chapter, the conclusions of the Project have been included, where the main ideas developed throughout the different chapters have been compiled. In addition, future recommendations for the Model have also been included, in order to ensure its viability in the future and to incorporate the use of new technologies.

## Contents

|  |    |
|--|----|
| 1. Introducción a la tecnología y fundamentos .....  | 8  |
| 1.1 La tecnología blockchain .....   | 8  |
| 1.1.1 Características principales .....  | 9  |
| 1.1.2 Tipos de blockchains .....   | 10 |
| 1.2 Ethereum .....   | 10 |
| 1.2.1 dApps dentro de Ethereum .....   | 11 |
| 1.2.2 Solidity .....   | 11 |
| 1.2.3 Prueba de participación .....  | 12 |
| 1.2.4 Tokens no fungibles .....  | 13 |
| 1.3 Smart contracts .....  | 13 |
| 1.3.1 Funcionamiento de los smart contracts .....  | 14 |
| 1.3.2 Aplicaciones actuales en las distintas industrias .....  | 15 |
| 1.4 Propiedad industrial, patentes y acuerdos de licencias cruzadas .....                            | 17 |
| 1.4.1 Patentes .....   | 17 |
| 1.4.2 Acuerdos de licencias cruzadas .....   | 18 |
| 1.5 Conclusiones .....   | 19 |
| 2. Análisis de la literatura .....   | 20 |
| 2.1 Bloqueo en la innovación y el desarrollo .....   | 20 |
| 2.2 Licencias cruzadas como solución al problema .....   | 22 |
| 2.2.1 Intel y Advanced Micro Devices .....   | 22 |
| 2.3 Principales soluciones que aportan contratos inteligentes en el sistema de patentes actual ..... | 23 |
| 2.4 Modelos de aplicación .....  | 24 |
| 2.4.1 Tecnología blockchain aplicada a la propiedad industrial .....                                 | 25 |
| 2.4.2 Tecnología blockchain aplicada a la cadena de suministros .....                                | 30 |
| 2.4.3 Tecnología blockchain aplicada a la medicina .....   | 31 |
| 2.5 Conclusiones .....   | 32 |
| 3. Modelo teórico y matemático .....   | 33 |
| 3.1 Condiciones de contorno .....  | 34 |
| 3.1.1 Gestión de la información .....  | 34 |
| 3.1.2 Sistema de pagos .....   | 35 |

|  |     |
|--|-----|
| 3.1.3 Interacción con los contratos inteligentes .....                         | 35  |
| 3.1.4 Tipos de nodos o validadores del sistema .....                           | 36  |
| 3.1.5 Territorialidad de las patentes.....                                     | 36  |
| Cálculo de los precios de las patentes.....                                    | 37  |
| 3.2 Estructura y funciones del modelo.....                                     | 43  |
| 3.2.1 Solicitar la explotación de una patente.....                             | 44  |
| 3.2.2 Acuerdo de licencias cruzadas.....                                       | 48  |
| 3.2.3 Creación de una nueva patente.....                                       | 51  |
| 3.3 Conclusiones .....   | 54  |
| 4. Aplicaciones prácticas del Modelo .....                                     | 56  |
| 4.1 Explicación previa de los valores establecidos de inicio.....              | 56  |
| 4.2 Caso Practico de solicitud de la licencia de una nueva patente .....       | 57  |
| 4.3 Caso Práctico de solicitud de licencia de explotación de una patente ..... | 67  |
| 4.4 Casos Prácticos de acuerdos de licencias cruzadas.....                     | 76  |
| 4.2.1 Primer Caso Práctico de acuerdos de licencias cruzadas .....             | 77  |
| 4.2.2 Segundo Caso Práctico de acuerdos de licencias cruzadas .....            | 88  |
| 4.5 Conclusiones .....   | 92  |
| 5. Memoria económica del Proyecto .....  | 94  |
| 5.1 Planteamiento de los cálculos .....  | 94  |
| 5.2 Datos económicos del proyecto .....  | 95  |
| 5.3 Análisis del Gasto total .....   | 96  |
| 5.3.1 Aparatos y Equipos físicos y lógicos.....                                | 96  |
| 5.3.2 Costes de Desarrollo.....  | 96  |
| 5.3.4 Administración y mantenimiento .....                                     | 101 |
| 5.4 Conclusiones .....   | 102 |
| 6. Conclusiones.....   | 104 |
| 6.1 Conclusiones y Objetivos .....   | 104 |
| 6.2 Desarrollos futuros.....   | 107 |
| Bibliografía.....  | 109 |
| Anexo A.....   | 113 |
| Anexo B.....   | 116 |
| Anexo C.....   | 121 |



## Introducción

El desarrollo de las tecnologías en la actualidad está aumentando cada año a un ritmo cada vez mayor<sup>1</sup>. A la hora de apostar por la innovación y el desarrollo este proyecto se ha llevado a cabo con la intención de optimizar el Sistema de Patentes tradicional por medio de la tecnología blockchain y los contratos inteligentes.

La motivación principal detrás de este Proyecto ha sido tratar de explicar una de las muchas aplicaciones que tiene la tecnología blockchain en la actualidad, para así, mostrar el potencial de esta en aplicaciones cotidianas. Junto con el vertiginoso desarrollo de esta tecnología, también aparece su enorme capacidad de adaptación frente a futuros cambios al estar basado en un código programado, capaz de modificarse en cualquier momento.

## Descripción de las partes del proyecto

El presente Proyecto trata de analizar las opciones que aparecen al aplicar una nueva tecnología como es la blockchain al sistema de patentes tradicional. Con esto, se busca optimizar todos los procesos que refieren a las patentes para agilizar así los trámites y promover la innovación y el desarrollo.

En el Capítulo 1, se ha realizado una introducción teórica a la tecnología blockchain, así como a los términos relacionados con las patentes que se consideran cruciales para después comprender correctamente proyecto a lo largo del avance del mismo. Entre los temas tratados en el Capítulo 1 se encuentran los principios de la tecnología blockchain, el funcionamiento de los contratos inteligentes dentro de la red de Ethereum, los acuerdos de licencias cruzadas y el interés que hay detrás de estos a fin de impulsar la innovación y el desarrollo.

Con este capítulo, se ha tratado de establecer correctamente las bases teóricas sobre las que se edifica el proyecto, haciendo hincapié en la estructura y el funcionamiento de

---

<sup>1</sup> OEPM, 2022

una blockchain, un contrato inteligente y explicando brevemente Ethereum la red sobre la que se ejecutarían los contratos inteligentes del Modelo.

Después, en el Capítulo 2 se ha llevado a cabo el análisis de la literatura para ponernos en perspectiva respecto a cuál es el avance en las dos partes que se buscan unir en el Proyecto, la tecnología blockchain y el sistema de patentes actual. En este capítulo se han analizado las características con las que cuentan tres aplicaciones distintas de la tecnología blockchain en la actualidad.

De cada una de las tres aplicaciones blockchain incluidas en el capítulo se han recogido ideas, características y funcionamientos que después serán incluidas en nuestro modelo

En el Capítulo 3, se ha desarrollado el Modelo del Proyecto desde el punto de vista teórico y matemático. Este Modelo cuenta con tres funciones fundamentales en el sistema de patentes: Solicitar la explotación de una patente, Acuerdos de licencias cruzadas y Creación de una nueva patente. Cada una de estas tres funciones se ha explicado de manera similar, desarrollando de manera simultánea la parte teórica y matemática del modelo dejando claro en todo momento cuáles son las variables de entrada y salida de cada función por medio de esquemas y aclaraciones.

Para cada una de las funciones se ha elaborado un diagrama de flujo distinto, donde se detallan todas las variables y procesos que tienen lugar para facilitar así su comprensión. También, se ha elaborado el código en Solidity de cada una de las tres funciones, que se podrá leer detenidamente en los anexos A, B y C.

Además de agilizar los trámites necesarios a la hora de crear una nueva patente, solicitar una licencia de explotación o un acuerdo de licencias cruzadas, el presente Modelo incluye también una calculadora de precios de las patentes. Su función es estimar de manera objetiva el precio de una licencia de una patente en base a unos parámetros como el tipo de licencia, la duración de la misma u otros cuyo razonamiento lógico se explica en el capítulo.

En el Capítulo 4 se ha puesto en práctica el modelo desarrollado en el capítulo anterior. Para cada una de las tres funciones se han adaptado casos reales de solicitudes de licencias de explotación, acuerdo de licencias cruzadas o creaciones de nuevas patentes

a nuestro modelo. En todos los casos prácticos se ha seguido el mismo procedimiento, primero se adaptaban los datos o se preestablecían condiciones y después se continuaba con el desarrollo del Modelo, simulando cuál sería su funcionamiento con esos datos asociados a sus variables de entrada y salida en cada uno de los.

Dentro de las variables con las que trabaja el Modelo, existen algunos valores incluidos en estas que se escapan del alcance del Proyecto ya que haría falta de su puesta en marcha en la realidad para poder obtenerlos. Por ello, en todo momento se ha especificado qué variables han sido predeterminadas porque se escapan del alcance del Proyecto y cuáles se han tenido que adaptar. También se ha incluido cómo sería el funcionamiento de la calculadora de precios de patentes, así como el razonamiento lógico detrás de cada cálculo. Por otro lado, se han diseñado capturas de pantalla de la interfaz que le aparecerían al usuario en cada una de las tres funciones durante su uso.

Después, en el Capítulo 5, se ha desarrollado la memoria económica del proyecto para poder estimar de la forma más precisa posible cuánto sería el desembolso económico necesario para llevar a cabo el desarrollo del proyecto. En este capítulo realizado unas estimaciones que se han considerado oportunas para poder reflejar la inversión inicial y el mantenimiento de una aplicación blockchain con las características del Modelo.

A la hora de presentar los costes de la implementación del modelo, se han planteado los costes de inversión inicial (Desarrollo y Auditoría) y los correspondientes a unas revisiones trimestrales que aseguren el correcto funcionamiento del mismo. En todo momento, para realizar estas estimaciones, se ha tenido en cuenta costes de proyectos con características similares al nuestro.

Por último, en el Capítulo 6, se han elaborado unas conclusiones del proyecto donde se recoge de manera general la información más relevante obtenida durante el desarrollo de las distintas partes del Proyecto, así como los objetivos cumplidos con el mismo. Además, también se incluirán recomendaciones futuras argumentadas según los datos obtenidos con el Modelo y a fin de asegurar la viabilidad de la aplicación de la tecnología blockchain al Sistema de Patentes Tradicional en el futuro.

Cada capítulo contará con una estructura similar, en primer lugar, una breve introducción, después el desarrollo del propio capítulo y por último unas breves conclusiones presentes en todos ellos. Estas conclusiones tratarán de sintetizar la información recogida en cada capítulo, así como las ideas que se han considerado más relevantes y sobre las que después se desarrollarán los siguientes capítulos.

## **Adecuación a los Objetivos de Desarrollo Sostenible**

El presente proyecto se ajusta a los objetivos del milenio de “Industria, innovación e Infraestructura” y “Ciudades y Comunidades Sostenibles”. Por medio de los avances tecnológicos que nos permiten la aplicación de la tecnología blockchain que antes se pensaban impensables, se ha tratado de innovar y optimizar la infraestructura detrás del sistema tradicional de patentes.



*Figura 1: Objetivos de desarrollo sostenible, Industria, Innovación e Infraestructura. ONU. c.2023*

Con el modelo propuesto en el proyecto, se promueve la innovación y el desarrollo de las empresas, facilitando que estas puedan obtener las licencias de patentes de la forma más sencilla y rápida posible.

El simple hecho de facilitar este procedimiento mediante las nuevas tecnologías estaría levantando antiguas barreras que frenan el desarrollo y la evolución de los nuevos procedimientos basados en tecnologías que aparecen a diario

Por otro lado, mediante la automatización se buscará el ahorro desde el ámbito económico, energético y temporal. Tratando así de promover el desarrollo de ciudades y comunidades sostenibles que sean capaces de adaptarse a las nuevas tecnologías a medida que estas avancen.



*Figura 2: Objetivos de desarrollo sostenible, Ciudades y Comunidades Sostenibles. ONU. c.2023*

# 1. Introducción a la tecnología y fundamentos

Se considera de vital importancia para la correcta comprensión del Proyecto realizar en primer lugar una breve introducción teórica de la tecnología blockchain y los smart contracts así como los conceptos de las patentes que estén relacionados con el tema del Proyecto, éste será el objeto del presente capítulo.

## 1.1 La tecnología blockchain

La tecnología blockchain y los programas que se desarrollan en ella, también llamados smart contracts, están siendo aplicados cada vez más a menudo en procesos que requieren de transparencia y certificación absoluta<sup>2</sup>.

Actualmente el desarrollo de esta tecnología ha seguido un ritmo exponencial, gran parte debido a la popularidad de las criptomonedas. Dejando a un lado a estas, nos centraremos en definir los principios de la tecnología blockchain y en la explicación de los conceptos necesarios para poder seguir las ideas que se planteen en este Proyecto.

La tecnología blockchain o tecnología de registro distribuido fue introducido en 2008 en el whitepaper de Bitcoin, donde se les permitía a los usuarios generar transacciones sin necesidad de intermediarios como podrían ser los bancos<sup>3</sup>. Cada una de las distintas redes utiliza un algoritmo de encriptación distinto, el cual permite que las transacciones solo puedan ser descifradas y por lo tanto leídas por aquellos que tengan en su posesión las claves o llaves necesarias.

---

<sup>2</sup> Bamakan, Nezhadsistani, Bodaghi, Qu, 2022

<sup>3</sup> Marchesi, Tonelli, Lunesu, 2022

### 1.1.1 Características principales

Las características principales que conforman un sistema basado en la tecnología blockchain son las siguientes<sup>4</sup>:

- Trazabilidad: todas las transacciones pueden ser rastreadas, conociendo así tanto el destinatario como desde dónde es enviada.
- Seguridad: el usuario dueño de una determinada dirección está asegurado por la criptografía de la red.
- Distribución: la información de la red no está solo en un organismo central sino en los distintos ordenadores que ejecutan la red.
- Inmutabilidad: los datos aceptados por la red no pueden ser alterados de ninguna forma.
- Descentralización: las transacciones se gestionan sin necesidad de una autoridad central.
- Programabilidad: por medio de los smart contracts se pueden programar acciones complejas para que luego sean realizadas por la red de forma automática.
- Económico: el sistema se gestiona de manera que mediante votaciones cualquiera podría llegar a modificar sus características, por lo que el coste de mantenimiento es nulo. También dependerá del nivel de saturación de la red.

Todos los aspectos anteriormente mencionados son de vital importancia para el futuro modelo que se planteará más adelante en este Proyecto porque se tendrán en cuenta para aplicarlos como cimientos para la estructura general del modelo.

---

<sup>4</sup> Marchesi, Tonelli, Lunesu, 2022

### 1.1.2 Tipos de blockchains

Por lo general las distintas blockchains se pueden diferenciar por cómo aceptan nuevos validadores y por cómo conceden el permiso de leer o escribir en ella<sup>5</sup>. Las blockchains también pueden ser abiertas (cualquiera puede acceder a ella y leer toda su información) o restringidas (solo determinados usuarios pueden acceder a ella)<sup>6</sup>.

En el caso de las blockchains abiertas, un ejemplo sería una empresa alimenticia la cual está interesada en que todos sus clientes sepan cual es la procedencia de los alimentos que consumen. Por otro lado, para el ejemplo de restringidas, podría darse el caso de una red de empresas farmacéuticas, la cual quiera controlar todos los envíos entre ellas y para ello utilice la tecnología blockchain.

Una de las aplicaciones de la tecnología blockchain que más llaman a la puerta de la innovación y el desarrollo es aplicada a la propiedad industrial<sup>7</sup>. Por medio de los smartcontracs se podrán automatizar gestiones que hoy en día requieren una gran inversión de tiempo y dinero. De todo esto se hablará más adelante cuando se hayan definido correctamente los fundamentos de la PI.

## 1.2 Ethereum

Es importante diferenciar también sobre que red será desarrollado el Proyecto, porque cada una tiene sus características, en este caso, Ethereum ha sido la seleccionada por las posibilidades que brinda al usuario en relación con los contratos inteligentes.

Ethereum es una red descentralizada cuyo token nativo es el ether<sup>8</sup>. Este token permite a los usuarios de esta interactuar con las distintas opciones que tiene la red, ya sea realizar transacciones, interactuar con smart contracts o con dApps<sup>9</sup>. En este Proyecto

---

<sup>5</sup> Antonopoulos, 2018

<sup>6</sup> Marchesi, Tonelli, Lunesu, 2022

<sup>7</sup> Cointelegraph, c. 2023

<sup>8</sup> Antonopoulos, 2018

<sup>9</sup> Ethereum.org, c. 2023



principalmente nos centraremos en los contratos inteligentes y en todas sus opciones. Es importante destacar que para el uso de los smart contracts será necesario también lidiar con el token nativo de la red y en algunas ocasiones con las dApps.

### 1.2.1 dApps dentro de Ethereum

Es importante entender cómo son las aplicaciones o interfaces de las páginas web que trabajan sobre la cadena de bloques de Ethereum, ya que estás servirán de ejemplo más adelante con el modelo que se planteará.

Las dApps son aplicaciones descentralizadas que no solo incluyen a los smart contracts, sino también datos fuera de la cadena de bloques como podría ser la interfaz de una página web que utiliza la tecnología blockchain<sup>10</sup>.

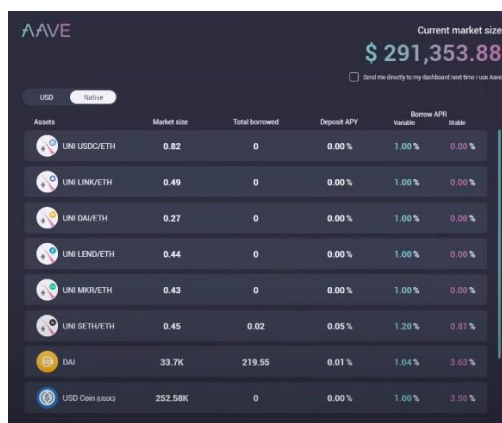


Figura 3: dApp de Aave, Fuente: Protocolo Aave, 2023

La anterior imagen muestra una dApp cómo es Aave, un protocolo para aportar liquidez de manera descentralizada comúnmente utilizado.

### 1.2.2 Solidity

En la red de Ethereum todo está escrito en el lenguaje de programación conocido como solidity<sup>11</sup>. Es un lenguaje de alto nivel que se utiliza para programar smart contracts,

<sup>10</sup> Ethereum.org, c. 2023

<sup>11</sup> Ethereum.org, c. 2023

este lenguaje comparte algunas características con JavaScript, C++ y Python<sup>12</sup>. Es un lenguaje de fácil<sup>13</sup> lectura ya que entre sus líneas de código incluye palabras como “contract” o “function”, que ayudan al lector a comprender el contrato.

Una de las cualidades más interesantes de solidity es la capacidad de trabajo que tiene con estructuras de datos complejas, lo que permite almacenar una gran cantidad de información de forma reducida en la red de Ethereum<sup>14</sup>. También incluye funciones de encriptación como firmas digitales.

Solidity utiliza un compilador para convertir los datos del lenguaje común a una programación de alto nivel que después será procesado por la Ethereum Virtual Machine (EVM)<sup>15</sup>, un pequeño ordenador que se ubica en cada nodo de la red. Este lenguaje de programación es Turing Complete<sup>16</sup>, lo que significa que en el mismo código se pueden incluir limitaciones y reglas que directamente serán incluidas en el contrato que a su vez irá incluido en la cadena de bloques. Esto último implica que será totalmente inmutable frente a cambios al estar incluido en una blockchain.

### **1.2.3 Prueba de participación**

Dentro los tipos de validaciones de red, los más comunes son la prueba de participación o la prueba de trabajo<sup>17</sup>, estos se han expandido entre las distintas cadenas de bloques que funcionan en la actualidad.

A la hora de validar las transacciones, la red de Ethereum funciona con el sistema de Proof of stake o prueba de participación. Por lo tanto, a diferencia de la otra gran blockchain de bitcoin, le permite funcionar sin que sea requerido un consumo energético tan grande como en su competidora. Además, el POS también nos permite encaminarnos hacia un futuro más sostenible respaldado por las nuevas tecnologías.

---

<sup>12</sup> Ethereum.org, c.2023

<sup>13</sup> Cointelegraph, 2022

<sup>14</sup> Antonopoulos, 2018

<sup>15</sup> Antonopoulos, 2018

<sup>16</sup> Ethereum.org, c. 2023

<sup>17</sup> Marchesi, Tonelli, Lunesu, 2022

En el sistema de validación de POS los usuarios bloquean sus criptomonedas (ether) en un contrato inteligente, creando así un nodo con una cierta cantidad bloqueada. Estos nodos son los encargados de, mediante un algoritmo de validación, asegurar la red y las transacciones que ocurren en ella<sup>18</sup>.

### **1.2.4 Tokens no fungibles**

Una idea que más tarde se desarrollará consiste en utilizar los tokens no fungibles como representantes de patentes sobre la cadena de bloques. Por lo tanto, es importante definirlos y establecer sus bases de funcionamiento para así después comprender correctamente el modelo propuesto.

Un token no fungible o NFT es un activo criptográfico el cual representa un objeto digital único e intangible con la información que este incluye. En la actualidad los NFTs se han utilizado para tokenizar (token como representante de algo fuera de la cadena de bloques) obras de arte, propiedades o datos, que de esta manera se incluyen y almacenan en la cadena de bloques con todos los beneficios que esto conlleva.

La red pionera en su implementación fue Ethereum con su token ERC-721. En este Proyecto nos centraremos en el tipo de token ERC-1155, que permite almacenar distintos tipos de datos en los contratos que lo admiten junto con la posibilidad de modificarlos.

## **1.3 Smart contracts**

Los smart contracts o contratos inteligentes son acuerdos automatizados entre el creador del contrato y el sujeto que lo recibe. Estos smart contracts están escritos en código y son ejecutados sobre una blockchain, haciéndolos inmutables e irreversibles<sup>19</sup>. Al ser automáticos, estos contratos eliminan la necesidad de un tercero que ejerza como intermediario en la ejecución de este. Por otro lado, también pueden activarse cuando

---

<sup>18</sup> Ethereum.org, c. 2023

<sup>19</sup> Antonopoulos, 2018

determinadas condiciones se den, siempre y cuando previamente se hayan programado correctamente<sup>20</sup>.

De esta forma, lo que se busca con estos contratos inteligentes es lo siguiente:

- Actuar de forma descentralizada: Al ser ejecutado sobre la una red blockchain, todo su funcionamiento y sus datos están siendo almacenados de forma descentralizada. Esto reduce las posibilidades de un fallo, ya que no sólo depende de un dispositivo que lo ejecute sino de tantos como validadores tenga la red<sup>21</sup>.
- Eliminar la necesidad de un intermediario y automatizar los procesos: al estar constituido por código de programación, se deja de depender de un organismo que valide la ejecución de estos, ahorrando así en tiempo y dinero<sup>22</sup>.

### **1.3.1 Funcionamiento de los smart contracts**

Para explicar el funcionamiento de los smart contracts se utilizará el siguiente ejemplo:

Una empresa A formaliza que cuando sus ingresos lleguen a diez mil euros, le pagará un porcentaje a la empresa B por sus servicios prestados. Una vez elaborado el código que define el contrato, ambas empresas, en caso de estar ambos de acuerdo, aceptarán y firmarán el contrato inteligente. Asumiendo que este contrato esté bien elaborado, esta condición de pago se habrá inscrito sobre la cadena de bloques, haciendo de ella misma inmutable frente a cambios y respaldada por la seguridad de la cadena de bloques sobre la que se ha ejecutado<sup>23</sup>.

En caso de que se cumpla la condición del contrato, la empresa A pagaría su debido porcentaje previamente pactado a la empresa B. El contrato inteligente debería incluir

---

<sup>20</sup> Huang, 2021

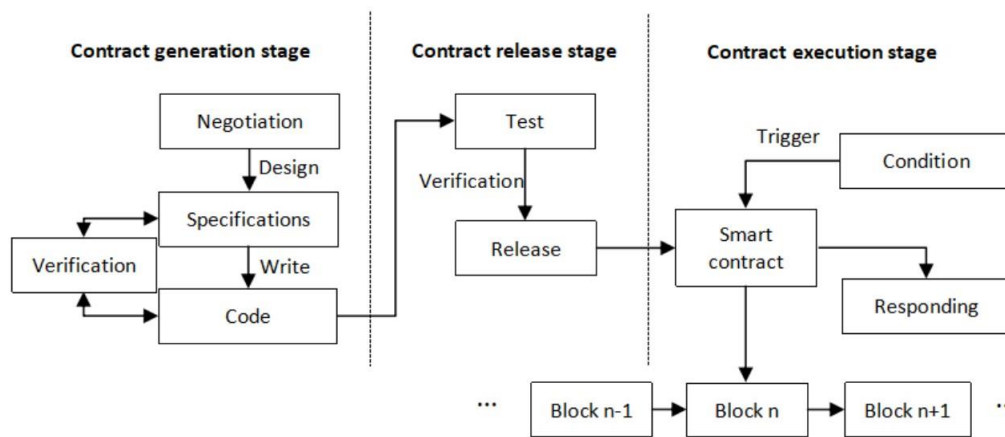
<sup>21</sup> Cointelegraph, c. 2023

<sup>22</sup> Cointelegraph, c. 2023

<sup>23</sup> Huang, 2021

todo tipo de cláusulas, así como constaría en un contrato tradicional. Es importante destacar también que esta gestión se habría llevado a cabo sin necesidad de un tercero que revise, acepte y ejecute la transacción.

El siguiente esquema representa de manera precisa todo el proceso anteriormente mencionado. En primer lugar, se establecen las bases del contrato, después se prueba su correcto funcionamiento y por último en caso de darse la condición que lo active, este actuaría y todo quedaría grabado sobre la cadena de bloques.



*Figura 4: Esquema de funcionamiento de un smart contract. Fuente: Research and application of Smart Contract Based on Ethereum Blockchain,2021.*

### 1.3.2 Aplicaciones actuales en las distintas industrias

Actualmente hay determinadas empresas que cuentan con la tecnología blockchain entre sus recursos de funcionamiento. Dentro las industrias a las que pertenecen estas empresas se encuentran la industria inmobiliaria, de seguros, identidad digital y centradas en cadenas de suministros<sup>24</sup>.

- Inmobiliaria: el proceso de compraventa de inmuebles en la actualidad requiere de múltiples trámites burocráticos los cuales consumen una gran cantidad de recursos. Si por medio de la red de Ethereum, un inmueble es tokenizado

<sup>24</sup> Cointelegraph, c. 2023

(concepto que se desarrollará más adelante), gran parte de los procesos de compraventa se automatizarían al estar todos inscritos en la blockchain.

- Seguros: el contrato estará subido a la red y la persona tomadora del seguro, en caso de querer contratarlo, podría firmarlo y lo tendría siempre subido en la cadena de bloques para consultar cualquier apartado. Además, toda la gestión de documentos donde se valida si es apto para contratar el seguro se realizaría de manera independiente y de forma casi instantánea ya que no habría que presentarse en ninguna oficina para entregar los documentos, todo se validaría de forma remota. La siguiente parte de financiación del contrato se realizaría de manera prácticamente instantánea a través de una transacción por la red.
- Cadena de suministros: con la ayuda de los contratos inteligentes, las distintas empresas podrían establecer puntos de reconocimiento a lo largo del proceso de envío para así comprobar que la mercancía llega a su lugar correctamente. Además, toda esta información estaría registrado en la cadena de bloques donde uno podría saber si ha habido algún error de entrega durante el trayecto.
- Identidad Digital: en la actualidad la información de cada usuario es de vital importancia para determinadas compañías. Si toda la información relativa a una determinada persona (datos médicos, DNI, pasaporte) se almacenan como token, esto ahorraría tiempo a la hora de realizar trámites. Por otro lado, estos datos podrían ser confidenciales, en ese caso permanecerían así estando respaldados por la seguridad de la tecnología blockchain.

Más adelante se recopilarán las ideas más importantes que todas estas industrias anteriormente mencionadas recogen de los smart contracts para después aplicarlas en nuestro modelo.

## 1.4 Propiedad industrial, patentes y acuerdos de licencias cruzadas

Previo a un análisis de los distintos artículos obtenidos para la fundamentación informativa del Proyecto, es necesario establecer unos conceptos relacionados con la propiedad industrial y algunos de los acuerdos que esta incluye.

En primer lugar, la propiedad industrial permite obtener unos derechos de exclusividad sobre determinadas creaciones que reciben una protección como derechos de propiedad<sup>25</sup>. La propiedad industrial cubre los siguientes derechos de propiedad industrial<sup>26</sup>:

- Diseños industriales: más centradas en la apariencia el producto
- Patentes y modelos de utilidad: protegen procesos de fabricación frente a la reproducción de estos.
- Topografías de semiconductores: protegen las distintas partes que componen un circuito integrado y la conexión entre ellas
- Marcas y nombres comerciales: centradas en proteger la reproducción de combinaciones gráficas que se diferencien a empresas en el mercado.

Es importante definir estas áreas de trabajo porque más adelante se aplicarán a casos actuales y se entrará más en profundidad.

### 1.4.1 Patentes

Una patente es un derecho exclusivo que se le concede a una persona por una invención determinada, que suele ser una solución tecnológica o una nueva técnica de realización de un proceso<sup>27</sup>. Para obtener esta patente se han de cumplir unos requisitos además de realizar unos determinados pasos para la creación de esta.

---

<sup>25</sup> OEPM, C.2023

<sup>26</sup> OMPI, c. 2023

<sup>27</sup> OMPI, c. 2023

En caso de que todos los requisitos se cumplan, el proceso para la creación de una nueva patente en la actualidad es tedioso<sup>28</sup>, ya que requiere de varios trámites y cuenta también con un gasto económico, el cual, más adelante, se desglosará para poder analizar correctamente el cambio que supone el modelo aplicado en este Proyecto.

En caso de querer explotar una patente, se deberá llegar a un acuerdo con el licenciante para permitir la explotación de esta. Este acuerdo suele terminar con una retribución económica por parte del licenciario hacia el licenciante. En caso de ser muchas patentes las que se necesitan el hecho de tener que tramitar cada proceso de explotación por separado puede elevar drásticamente los costes. Frente a este problema anteriormente mencionado hay distintas opciones como podrían ser los consorcios de patentes o los acuerdos de licencias cruzadas entre empresas.

### **1.4.2 Acuerdos de licencias cruzadas**

A la hora de crear comercializar nuevos productos, la empresa interesada debe tener la licencia de todos los componentes que intervienen en el proceso de creación de este. Esto, en determinadas situaciones, dependiendo de la industria puede ser un camino lleno de obstáculos y gastos de administrativos<sup>29</sup>. Es aquí donde los acuerdos de licencias cruzadas entre empresa son una opción interesante para aplicar<sup>30</sup>.

Las licencias cruzadas entre dos empresas se acuerdan de manera conjunta cuando cada una de ellas cuenta con una o más patentes de la otra empresa en su respectivo proceso de fabricación. Es un interés mutuo por no incumplir con ninguna normativa de patentes, lo cual les haría pagar la correspondiente sanción económica y en algunos casos hasta detener el proceso de fabricación<sup>31</sup>.

Ahora ambas empresas son libres de crear su producto sin el riesgo de incumplir ninguna normativa referida al uso ilegal de patentes. Es importante destacar que

---

<sup>28</sup> Shapiro, 2001

<sup>29</sup> Shapiro, 2001

<sup>30</sup> Shapiro, 2001

<sup>31</sup> Sánchez García, 2018



solamente las dos empresas implicadas que forman el acuerdo tendrían permitido el libre uso de estas patentes<sup>32</sup>.

La industria tecnológica y la farmacéutica son las dos industrias que más cuentan con este tipo de licencias ya que a la hora de invertir en I&D, este tipo de acuerdos abarata mucho los costes aun teniendo que pagar una cierta cantidad económica en determinadas ocasiones<sup>33</sup>. Las licencias cruzadas agilizan el proceso de comercialización al no ser necesario gestionar todos los permisos para cada una de las patentes necesarias de un producto<sup>34</sup>.

## 1.5 Conclusiones

Estos acuerdos de licencias cruzadas son la solución a muchos de los problemas con los que cuentan las empresas a la hora de comercializar un nuevo producto. Aun así, estos acuerdos siguen requiriendo una inversión de tiempo y dinero que podría ser reducida si se aplicara la tecnología blockchain al sistema de patentes actual.

Por lo tanto, para concluir, una vez han sido asentados los conceptos más importantes de la tecnología blockchain, el presente proyecto se basará en su totalidad en la red de Ethereum. Esta red cuenta con las características principales de una blockchain abierta, también es importante aclarar que los NFTs junto con las dApps servirán en su conjunto como puente entre la tecnología blockchain y el sistema tradicional, todo ello administrado con smart contracts. En el siguiente capítulo se analizarán las distintas opciones que han sido recogidas durante el proceso informativo para aplicar toda esta tecnología a la Propiedad Industrial.

---

<sup>32</sup> Sidak, 2015

<sup>33</sup> Shapiro, 2001

<sup>34</sup> Giuri, Torrisi, 2010

## 2. Análisis de la literatura

Una vez se han asentado los conocimientos tecnológicos y teóricos que serán constantemente referenciados a lo largo del proyecto, en el presente capítulo se procede a analizar y comentar los distintos artículos que han sido recopilados durante la parte informativa del Proyecto. Todos estos artículos contienen ideas prácticas que servirán como cimientos para el desarrollo del futuro modelo.

### 2.1 Bloqueo en la innovación y el desarrollo

El proceso de creación y comercialización de un nuevo producto en la actualidad cuenta con multitud de obstáculos y trámites legales que dificultan el lanzamiento del producto<sup>35</sup>. Ciertamente existen distintas alternativas<sup>36</sup> que facilitan el proceso legal de obtener todas las patentes necesarias, como podrían ser los acuerdos de licencias cruzadas entre empresas o los consorcios de patentes.

Aun así, se está produciendo un efecto parecido al del cuello de botella con las patentes, donde no se facilita la comercialización de nuevos productos y se está frenando la innovación debido a todos los procesos que son necesarios a la hora de crear una nueva patente o tramitar una solicitud para la explotación de una ya existente<sup>37</sup>.

El hecho de que se estén creando tantas patentes en la actualidad implica que es realmente difícil conseguir un producto que no infrinja múltiples derechos de patentes<sup>38</sup>, esto último sugiere que determinada información sobre una patente (fecha de emisión, técnicas o productos que protege) cobra especial importancia para tener en cuenta en un futuro.

Por ejemplo, la fecha de emisión de las patentes es de vital importancia<sup>39</sup> frente a la de la creación de un nuevo producto. Si se diera el caso en el que se fuese a crear un

---

<sup>35</sup> Shapiro, 2001

<sup>36</sup> Giuri, Torrisi, 2010

<sup>37</sup> Shapiro, 2001

<sup>38</sup> Giuri, Torrisi

<sup>39</sup> Shapiro, 2001

producto que utilizara un método ya patentado, habría una retribución económica para el licenciante que, en la mayoría de los casos, zanjaría la situación<sup>40</sup>. El problema aparecería cuando, estando un producto ya en circulación y en constante fabricación, se emita una patente que incluya en si misma algún método utilizado en la fabricación del producto. En ese caso el licenciante podría solicitar medidas cautelares<sup>41</sup>, amenazando así con el detenimiento de la producción de la empresa afectada.

La situación previamente comentada refleja lo importante que es la tenencia de todas las patentes de un producto a la hora de su creación y comercialización. En industrias como la tecnológica, las empresas pueden verse inmersas en una situación comprometida relacionada con las patentes por el simple hecho de desconocer de la existencia de una patente, o por no haber patentado un diseño determinado (aun habiendo tenido los conocimientos para ello) en el pasado<sup>42</sup>.

Por otro lado, Cournot (1838) nos demuestra mediante su modelo económico, que, tanto para la economía de una industria determinada como para una empresa, lo más beneficioso es que se llegara a un acuerdo como podría ser un consorcio de patentes o un acuerdo de licencias cruzadas para la gestión de estas.

Se explicará la idea anteriormente mencionada con el siguiente ejemplo:

Para que una empresa pueda comercializar legalmente un producto, debe tener la licencia de explotación de tantas patentes (llamemos al número N) como procesos, técnicas o productos ya patentados aparezcan en su proceso de fabricación. Si cada una de esas patentes perteneciera a una empresa distinta, la empresa en cuestión debería realizar el proceso de negociación de explotación de una patente con N empresas.

Cada una de estas gestiones no consisten simplemente en una retribución económica al licenciante (que también), pero además habría que incluir el gasto económico relacionado a la administración de la licencia y el respectivo consumo de tiempo. El

---

<sup>40</sup> Rose, 2020

<sup>41</sup> Rose, 2020

<sup>42</sup> Shapiro, 2001

precio total de la obtención de las N patentes sería mayor<sup>43</sup> que si estas se agruparan o si se llegara a acuerdos de licencias cruzadas.

Entonces, el hecho de agrupar y simplificar la obtención de licencias es beneficioso para el sistema<sup>44</sup>, ya que incentiva a la creación e innovación, reduciendo el número de obstáculos a los que se enfrenta una empresa con intención de obtener una licencia de patentes.

## **2.2 Licencias cruzadas como solución al problema**

En un acuerdo de licencias cruzadas, ambas empresas se benefician<sup>45</sup> de la libertad de fabricación de la que disponen. Además, no es necesario que se pague una tasa por unidad como sí es el caso en una negociación habitual de explotación de patentes. Estos acuerdos incluyen dentro de sus negociaciones<sup>46</sup> las patentes emitidas, pendientes de emitir, y el permiso de explotación por un tiempo definido entre la fecha del acuerdo hasta una fecha previamente pactada.

Los acuerdos de licencias cruzadas o ALC (se referirá a los mismos así por simplificación de la lectura) son la norma general<sup>47</sup> en los mercados donde el diseño y la fabricación de microprocesadores es la actividad principal de estos.

### **2.2.1 Intel y Advanced Micro Devices**

Uno de los famosos casos de acuerdos de licencias cruzadas entre empresas de microprocesadores es el de Intel y Advanced Micro Devices (AMD)<sup>48</sup>. Se explicará el caso brevemente, para después analizar sutilmente el resultado.

Este ALC les permitía a ambas empresas acceder a la tecnología ya desarrollada de la otra, la primera idea que viene a la mente de uno es que, si Intel por ejemplo lanzase un

---

<sup>43</sup> Jeon, Lefouili, 2018

<sup>44</sup> Shapiro, 2001

<sup>45</sup> Sánchez García, 2020

<sup>46</sup> Sánchez García, 2020

<sup>47</sup> Shapiro, 2001

<sup>48</sup> Siuta, 2019

producto, AMD al tener acceso a ese producto ya terminado y todo su proceso de fabricación podría copiarlo y lanzarlo ellos mismos. El problema es que en un mercado como es el de los microprocesadores, la tecnología y los nuevos productos avanzan a pasos agigantados, y ninguna de las dos empresas se podría permitir el lujo de esperar a que la otra sacara un nuevo producto para después copiarlo y lanzarlo ellos mismos.

Los microprocesadores de AMD consiguen un rendimiento ligeramente inferior al de los Intel, pero es esa pequeña diferencia la que resulta en que Intel retenga el 90% del mercado de los microprocesadores.

Entonces, uno puede pensar que realmente este acuerdo de licencias cruzadas no es beneficioso para AMD porque no consiguen atraer una cantidad de clientes significativa. No obstante, este acuerdo aprueba el uso de la tecnología Intel por parte de AMD para determinados sectores del mercado en los que Intel no es un rival, como podría ser la fabricación de tarjetas gráficas. En este mercado gracias a la tecnología ya desarrollada por Intel, AMD ha conseguido enfrentarse con resultados claramente positivos a su principal competidor Nvidia.

Refiriéndonos al caso anteriormente planteado, se puede observar que los ALC no sólo benefician a las dos empresas implicadas en el aspecto de ahorro de trámites burocráticos que consumen tiempo y dinero, sino que, también las transforma<sup>49</sup> en empresas más competitivas en sus industrias al contar con un porfolio de patentes notablemente más amplio que el que tendrían ellas por separado.

## **2.3 Principales soluciones que aportan contratos inteligentes en el sistema de patentes actual.**

Para agilizar todos los procesos con los que cuenta el sistema de patentes actual aun habiendo aplicado un acuerdo de licencias cruzadas, se plantea la introducción de la tecnología blockchain, más en concreto los smart contracts al sistema de patentes.

---

<sup>49</sup> Siuta, 2019

Actualmente el sistema de obtención de patentes cuenta con algunas cuestiones con margen de optimización, como podrían ser el tiempo de espera, las dificultades en el pago a los licenciarios y el coste monetario que conllevan. La tecnología blockchain y los smart contracts vendrían a solventar estos problemas con ciertas modificaciones.

A continuación, se presentan tres aspectos que podrían ser optimizados utilizando los contratos inteligentes. Más adelante se explicará una estructura para llevar a cabo ese proceso de optimización.

1. Tiempo de gestión: al llevarse a cabo de forma prácticamente automatizada, el tiempo sería mucho menor que el actual (8-15 meses). Aun así, no todo se podría gestionar de forma descentralizada sobre la cadena de bloques, ya que podría darse el caso en el cual haya que resolver alguna gestión de manera presencial, prescindiendo así del código<sup>50</sup>.
2. Coste monetario que conllevan: Actualmente conseguir una patente en España asciende a 1200<sup>51</sup> euros, sin contar con el tiempo que habría que invertir en esa gestión. Si le retiramos el sueldo de parte de esos trabajadores cuya función es aprobar y tramitar las creaciones de nuevas patentes el coste total se vería reducido. Esto se debe a que en gran parte el proceso estaría automatizado por un sistema basado en código, abaratando de forma significativa el trámite.
3. Automatizar el pago de las “royalties”: Al realizarse los pagos por medio de un sistema descentralizado y basado en código, estos se tramitarían de forma prácticamente instantánea y sin necesidad de intermediarios y su respectiva dependencia.

## 2.4 Modelos de aplicación

Para el correcto desarrollo del trabajo es importante también poner en práctica todas las opciones que aparecen al incorporar la tecnología blockchain a cualquier sistema de la

---

<sup>50</sup> Bodó, Gervais y Quintais, 2018

<sup>51</sup> OEPM, c. 2023

actualidad. En el presente proyecto, como ya se ha mencionado anteriormente, nos centraremos en aplicarlo al sistema de patentes actual.

Se analizarán y comentarán las ideas que se consideren más interesantes para el Proyecto de tres modelos distintos. Cada de ellos contará con sus distintas aplicaciones específicas, pero nos centraremos en aquellas más generales que después puedan ser modificadas en caso de ser necesario para después utilizarlas.

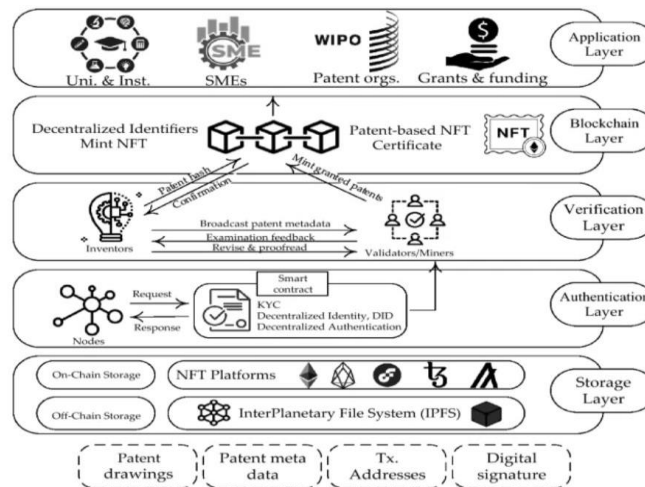
### **2.4.1 Tecnología blockchain aplicada a la propiedad industrial**

El siguiente modelo planteado se basa en la estructura que explican Motjaba, Bamakan, Nezhadsistani, Bodaghi y Qu, (2022) en “*Patents and intellectual property assets as non- fungible tokens; key technologies and challenges*”, todas las imágenes y referencias no incluidas del presente capítulo han sido obtenidas del artículo anteriormente mencionado.

#### **2.4.1.1 Ideas generales de la estructura aplicada**

Una de las propuestas para llevar a cabo esta implementación sería combinando smart contracts y NFTs. Por medio de un NFT se interactuaría con el smart contract encargado de gestionar una patente. Este token no fungible incluiría en sí mismo todos los datos relacionados con la nueva patente.

El modelo planteado se divide en 5 capas, almacenamiento, autenticación, verificación, blockchain y aplicación. La siguiente imagen muestra un esquema general de todas las capas.



**Figura 5: Estructura del modelo de aplicación de la tecnología blockchain a la propiedad industrial. Fuente: Patents and intellectual property assets as non- fungible tokens; key technologies and challenges, 2022**

### 2.4.1.2 Capa de almacenamiento

En primer lugar, se encuentra la capa de almacenamiento, esta funcionaría de forma completamente descentralizada, por lo que se ahorraría al optimizar el uso del espacio del almacenamiento. Por otro lado, al existir multitud de copias en distintos nodos, se evitarían cuellos de botella en servidores centrales agilizando las descargas y las subidas a la red.

La información de cada una de las patentes iría incluida en los metadatos de un NFT, de esta forma cada NFT representaría a una patente, abarcando toda a la información referida a la misma. El hecho de estar incluida en un NFT facilita el proceso a la hora de realizar gestiones como pago de royalties o traspasos de propiedad al poder interactuar con un contrato inteligente.

Los metadatos de ese NFT contienen información que lo define y diferencia del resto de tokens. A la hora de almacenar estos metadatos se puede hacer de dos formas, en la cadena de bloques o fuera de las cadenas de bloques. Para este Proyecto, se seleccionará el procedimiento el cual lo almacena en la blockchain teniendo en cuenta que la cantidad que se deberá pagar por transacciones al utilizar esta red dependerá de la saturación de esta.



### **2.4.1.3 Capa de autenticación**

En segundo lugar, tenemos la capa de autenticación, en esta capa, se utiliza el método de identidad descentralizada o DID que consiste en que el usuario en cuestión tendrá asociado a una dirección de la blockchain unos datos personales que después le servirán como método de identificación. Este proceso validará las credenciales del usuario para confirmar que es elegible para utilizar el sistema.

Gracias a esta capa sería posible seguir la pista de una patente a través de la cadena de bloques para conocer así quién es el licenciante. Este proceso de autenticación se conoce como KYC (Know your customer) y se utiliza en la mayoría de los protocolos que trabajan con la tecnología blockchain. Esta estructura de modelo permite un funcionamiento donde se pueda verificar la identidad del usuario de manera descentralizada y sin necesidad de un tercero. Este modelo genera smart contracts programados para gestionar licencias de patentes que se ejecutan sobre la cadena de bloques y permite enviar y recibir transacciones.

En la actualidad todo este proceso de autenticación es llevado a cabo por la Organización Mundial de la Propiedad Intelectual u otras empresas centralizadas, esta labor se ejecuta de manera tradicional llegando a ser ineficiente y costosa en alguno de los casos.

### **2.4.1.4 Capa de verificación**

En tercer lugar, se encuentra la capa de verificación, en blockchains restringidas solo determinados nodos pueden leer y escribir en la red. Este tipo de privilegios que son concedidos a los nodos, podrían servir para delegar en la red cuáles son los usuarios capaces de validar y conceder las licencias de patentes.

Para llevar a cabo todo este proceso de forma totalmente descentralizada, habría que digitalizar toda la información referente a una patente. Este proceso, aún tedioso, es necesario debido al estratosférico avance de las tecnologías en la actualidad. Mediante

una simple aplicación que actúe como un formulario tradicional se podría subir toda la información necesaria de la patente.

Una vez la información hubiera sido subida por el usuario a la red, el nodo validador emitiría una transacción para aceptar o denegar la emisión de esa patente y que esto quedase reflejado en la cadena de bloques.

Existe el riesgo de que, a la hora de subir una patente, el minero o nodo encargado de validar esa transacción la rechace y se apropie él mismo de la idea. Por esto mismo, el inventor deberá guardar de forma privada su patente mediante la prueba de existencia. Esto último consiste en subir la información a la cadena de bloques de manera privada, de tal forma que hasta que no se valide la transacción la información de la patente no se pueda visualizar de forma pública. Así el licenciante se aseguraría poder justificar su invención con la fecha que quedaría registrada en la cadena de bloques. Con este mismo procedimiento, se podrían ir también guardando el desarrollo de una patente de forma privada, para en todo momento tener pruebas de las distintas fases del proceso.

Para el correcto funcionamiento de esta estructura, los nodos encargados de validar la información referente a la patente deberán ser usuarios profesionalmente cualificados y aceptados por la respectiva oficina de patentes. Podrían recibir un certificado digital que esté incluido en la cadena de bloques y valide el acceso a la misma como usuario capacitado para leer y aceptar transacciones en la red.

#### **2.4.1.5 Capa blockchain**

En cuarto lugar, se encuentra la capa blockchain, el principal objetivo de esta es proveer a esta estructura de gestión de la propiedad industrial con todas las características de la tecnología blockchain para así mejorarla. A la hora de elegir la red, habría distintas opciones, pero como se ha comentado anteriormente en este Proyecto se elegirá la red de Ethereum debido a la flexibilidad que esta ofrece en cuanto a smart contracts.

Para que el sistema de patentes implemente los NFTs y la tecnología blockchain, se deberá seguir el siguiente modelo de funcionamiento:

- El creador deberá registrarse en una aplicación descentralizada (dApp) para patentar su creación, la información referente a su identidad será requerida.
- Se deberá subir a la red toda la información de su patente relacionada con la propiedad industrial. Al realizarse sobre la cadena de bloques se cuenta con la trazabilidad de esta y está protegida frente a copias o manipulaciones. En este momento, en caso de desearse así, la patente estaría a la vista de todos en la cadena de bloques.
- En caso de que un nuevo usuario quisiera acceder a la información de una patente, este en primer lugar, debería registrarse e identificarse en la dApp, una vez hecho eso, tendría que solicitar al licenciante el acceso. Después, si el licenciante aceptara, un smart contract sería el encargado de administrar esa gestión. Ambas partes acordarían unas condiciones, pudiendo ser estas un intercambio de licencia de patentes o una retribución económica por parte del licenciario al licenciante. Una vez los dos hayan accedido y firmado un acuerdo de no divulgación, se ejecutaría el contrato sobre la cadena de bloques haciendo de este inmutable frente a cambios.
- Los acuerdos entre empresas serían resueltos por aquellos nodos que están capacitados para leer y escribir sobre la cadena de bloques. Estos acuerdos incluyen traspasos de derechos de la propiedad industrial, pruebas de propiedad y múltiples disputas que podrían llegar a suceder.

#### **2.4.1.6 Capa de aplicación**

Por último, tendríamos la capa de aplicación, en ella se incluirían todas las instituciones, gobiernos, universidades o empresas que quieran formar parte de esta infraestructura para almacenar e intercambiar derechos de propiedad intelectual de manera descentralizada, segura y sin necesidad de trámites burocráticos tradicionales.

Cualquier usuario que esté de acuerdo con las condiciones del licenciente podrá pagar y de manera prácticamente instantánea tendrá acceso a toda la información relacionada con una patente, todo esto sin necesidad de que ambas partes tengan contacto directo para la negociación. En la actualidad las patentes están reguladas jurisdiccionalmente alrededor del mundo, con el sistema propuesto, se eliminarían las barreras geográficas con las que se cuenta en la actualidad. Además, el hecho de que el acceso a una licencia de una patente sea tan fácil tendría un gran impacto positivo en la innovación y el desarrollo potenciando la creación y el avance de las nuevas tecnologías.

#### **2.4.2 Tecnología blockchain aplicada a la cadena de suministros**

En segundo lugar, otro modelo analizado del cual se obtendrán ideas para después aplicarlas al nuestro es el de Marchesi, Tonelli y Lunesu, (2022) en “*A blockchain architecture for industrial applications*”. En él se plantea una estructura la cual podría servir para administrar una cadena de suministros alimenticios, donde en todo momento estará disponible la trazabilidad de cualquier producto y verificar la calidad de este.

Entre las características del modelo anteriormente mencionado, se encuentra la posibilidad de escanear un código QR<sup>52</sup> (que irá impreso en la etiqueta) y que éste te redireccione mediante tu buscador de internet predeterminado a la cadena de bloques. Después, una vez en la cadena de bloques, mediante una dApp (previamente creada para funcionar como puente entre la tecnología blockchain y todo lo que hay fuera de ella) que contará con una interfaz de fácil interpretación, el usuario será capaz de analizar la proveniencia de ese producto, y todos los eventos relacionados con su cadena de suministros<sup>53</sup>.

Los usuarios que compren esos productos tendrán acceso únicamente como lectores de la cadena de bloques. Por otro lado, personal cualificado será el encargado de validar la información que se encuentra en la blockchain y modificarlo pertinentemente en caso de que fuera necesario.

---

<sup>52</sup> Marchesi, Tonelli, Lunesu, 2022

<sup>53</sup> Marchesi, Tonelli, Lunesu, 2022

Gracias a este modelo se cumplirían objetivos como documentar toda la información relevante de la cadena de suministros de un producto de forma totalmente transparente y con las comodidades de los nuevos dispositivos móviles. Además, también se permitiría a los laboratorios y profesionales agrícolas aportar y avalar su determinado sello de calidad<sup>54</sup>.

Por último, el hecho de tener tanto la producción como la distribución controlada por medio de la tecnología blockchain, ayudaría a que no se introdujeran en el proceso de distribución nuevos productos de procedencia desconocida<sup>55</sup>, asegurando así la calidad del producto.

### **2.4.3 Tecnología blockchain aplicada a la medicina**

En tercer lugar, tenemos el modelo planteado por Lin, Zhang, Li, Jue y Sun, (2020), en *“A survey of application research based on blockchain smart contract”*. Una de las ideas que destacan en él es una estructura capaz de gestionar el sistema médico actual. En él, se podría almacenar toda la información referida a un determinado paciente.

Si esta blockchain sobre la que se ejecuta el sistema es de acceso restringido, cada paciente podría acceder a la misma como lector y ser capaz de ver sus registros médicos, recetas e incluso futuras citas<sup>56</sup>. Por otro lado, serían aquellos con un permiso especial debido a su formación profesional quienes podrían acceder tanto como modificadores o como lectores de la información.

Para poder privar del acceso a aquellos usuarios que no estén cualificados, en el presente modelo se propone una idea interesante para gestionar la situación. En lugar de asignar un NFT como título digital a cada uno, en este caso, al personal sanitario se le asignaría una dirección<sup>57</sup> en la cadena de bloques que más adelante se incluiría dentro del contrato inteligente como dirección válida para leer y modificar

---

<sup>54</sup> Marchesi, Tonelli, Lunesu, 2022

<sup>55</sup> Marchesi, Tonelli, Lunesu, 2022

<sup>56</sup> Lin, Zhang, Li, Jue y Sun, 2022

<sup>57</sup> Lin, Zhang, Li, Jue y Sun,

Al depender de la tecnología blockchain, esta información de cada paciente pasaría a estar respaldada por la misma y dejaría de depender del paciente como sí sería el caso en el sistema actual. Aunque la tarea de subir toda la información a la cadena de bloques fuera tediosa, en el largo plazo, debido a la cantidad de información que proporcionan las nuevas tecnologías medicas saldría claramente rentable<sup>58</sup>. Además, se eliminaría el riesgo de que se perdieran o filtrasen los datos médicos de algún paciente durante un traspaso de información<sup>59</sup>.

## 2.5 Conclusiones

Para concluir, tras observar las anteriores estructuras planteadas, se puede afirmar que la tecnología blockchain juntos con los smart contracts, por medio de la descentralización, la automatización de procesos y la seguridad de la propia tecnología, podrían solucionar los problemas de optimización que han sido expuestos.

Además, por otro lado, después de analizar el resultado del acuerdo de licencias cruzadas entre AMD e Intel, podemos concluir que realmente son la alternativa más atractiva para este tipo de disputas. Habiendo entendido y analizado casos más prácticos de la idea principal del proyecto, en el siguiente capítulo se elaborará un modelo a nivel teórico, el cual basará su funcionamiento en los procesos o ideas recogidas en el presente capítulo.

Todos los modelos analizados cuentan con características de interés para este proyecto, pero es el modelo planteado por Motjaba, Bamakan, Nezhadsistani, Bodaghi y Qu el que se considera con mayor adaptación hacia los conceptos tratados en el presente proyecto, ya que plantea soluciones por medio de los NFTs y los contratos inteligentes para los problemas de optimización en el sistema tradicional de patentes.

Por ello, para el futuro modelo se extraerán ideas de las tres estructuras planteadas, pero, será elaborado basándonos principalmente en las ideas introducidas por el modelo explicado que fundamenta su aplicación a la propiedad industrial

---

<sup>58</sup> Lin, Zhang, Li, Jue y Sun,

<sup>59</sup> Lin, Zhang, Li, Jue y Sun,

### 3. Modelo teórico y matemático

En el presente capítulo, se desarrollará el modelo del Proyecto, el cual permita automatizar funciones del sistema de patente actual. En primer lugar, se explicarán aquellas partes del modelo que, debido a su complejidad se escapan de la capacidad del proyecto, por lo que se fijarán las condiciones del funcionamiento del modelo.

Después se desarrollarán cada una de las tres funciones con las que cuenta el modelo por separado, explicando sus variables de entrada, salida y su funcionamiento intermedio. Por último, se expondrán unas conclusiones obtenidas tras haberlo planteado.

En primer lugar, se explicará de forma conceptual el modelo que ha sido desarrollado tras la recopilación informativa realizada en el capítulo anterior. Para llevar a cabo esta tarea se detallan las variables que han sido tratadas como entrada o salida, así como lo procesos intermedios que han tenido lugar.

Es importante, previo a comenzar el capítulo, establecer algunas partes del modelo que su desarrollo no corresponde en el actual proyecto, por lo que se describirán brevemente para así poder entender el modelo en su totalidad. Cabe destacar que, la interfaz de funcionamiento del modelo de cara al usuario será por medio de una dApp, la cual servirá como puente entre la tecnología blockchain, los smart contracts y el funcionamiento tradicional. Esto último se plantea gracias al modelo propuesto por Motjaba, Bamakan, Nezhadsistani, Bodaghi y Qu, (2022) en *“Patents and intellectual property assets as nonfungible tokens; key technologies and challenges”*.

## 3.1 Condiciones de contorno

A continuación, se explicarán brevemente las partes del modelo que no se desarrollan porque queda fuera de la capacidad del proyecto. Aun así, se considera importante definir las partes para que el modelo quede explicado prácticamente en su totalidad, entre estas partes se encuentran el tratamiento de la información, los pagos y las interacciones con los contratos inteligentes. Estas tres partes, englobarán los cinco apartados tratados a lo largo del subcapítulo.

Todo aquello que no se desarrolle, pero si se describe en este subcapítulo, servirá como condiciones de contorno para establecer las bases sobre las que este funcionará.

### 3.1.1 Gestión de la información

Gran parte del actual modelo se basa en la subida de información a la cadena de bloques, así como el envío de información relacionada con patentes que estará almacenada de distintas formas:

- Toda la información que refiere a una patente cuando se está elaborando una solicitud, irá incluida en un archivo de texto, el cual pertenecerá a un tensor donde se almacenarán las variables de entrada.
- Cuando se emita una patente, la información de esta irá incluida en los metadatos de un NFT. Este token no fungible habrá sido emitido por un contrato inteligente una vez se hayan cumplido unas determinadas funciones previamente establecidas, esta idea se basa en el planteamiento que proponen **Motjaba, Bamakan, Nezhadsistani, Bodaghi y Qu, (2022)** en *“Patents and intellectual property assets as nonfungible tokens; key technologies and challenges”*.

La información de los metadatos del NFT incluirá en su interior no solo las características de la patente en cuestión, si no también información referente a las transacciones en la cadena de bloques que se lleven a cabo debido a las cláusulas o royalties previamente acordados.



### **3.1.2 Sistema de pagos**

Otra de las cuestiones que agilizan el funcionamiento del sistema de patentes, sería el pago automático por medio de un smart contract, Ambas partes podrían acordar previamente el pago de los royalties por medio de las criptomonedas, al hacer eso, el tiempo requerido para que el dinero llegase a la otra parte, no dependería del día en el que se llevase a cabo o de la hora.

Como se ha mencionado anteriormente la red seleccionada para este modelo es la de Ethereum, por lo tanto, la criptomoneda transferida sería cualquier token que fuese ejecutado en la misma.

Los pagos por medio de la red de Ethereum contarían con una comisión extra proveniente de las tasas que cobra la red por su uso, la cantidad de esta dependería al 100% de lo saturada que estuviera la red en el momento de enviar la transacción o de ejecutar el contrato inteligente.

### **3.1.3 Interacción con los contratos inteligentes**

La mayor parte de las gestiones realizadas en el modelo se llevarían a cabo por medio de los contratos inteligente, por lo que consideran de vital importancia, explicar cómo serán estas interacciones.

En primer lugar, cada acuerdo que tenga lugar en las distintas funciones será gestionado por medio de un contrato inteligente. En este contrato irán descritas las condiciones previamente pactadas por cada una de las partes, así como las cláusulas por las que habría que pagar una penalización o rescindiría el contrato.

En segundo lugar, todas las empresas del sistema tendrán asociado un número de identificación que a su vez esté irá asociado a una dirección en la cadena de bloques. De esta forma se evita que alguien se haga pasar por una empresa, ya que, al no estar accediendo con la dirección de la otra empresa, no podría interactuar con el contrato inteligente como tal.

En tercer lugar, estos contratos inteligentes, también serán capaces de acuñar un NFT el cual sirva como representante criptográfico de una patente. Por último, para el correcto funcionamiento del sistema, hará falta que determinados usuarios ejerzan como validadores de información para agilizar y facilitar la interacción con los contratos inteligentes, estos usuarios serán los llamados validadores que se explicarán más adelante.

### **3.1.4 Tipos de nodos o validadores del sistema**

Es importante realizar esta diferenciación para la correcta comprensión del modelo, éste contara con dos tipos de nodos, unos validarán cualquier tipo de transacción en la red de Ethereum y otros, validarán la información que refiere al sistema de patentes.

Los nodos que validan las transacciones en la red de Ethereum no son propios del modelo, si no que irán variando según vaya cambiando el algoritmo de la red. Es decir, esos nodos no solo validaran las transacciones de nuestro modelo, si no que validaran todas las transacciones que tienen lugar en la red de Ethereum completa.

Por otro lado, tenemos nuestros validadores, que serán referidos así para simplificar la diferenciación. Estos son los encargados de validar la información de cada patente, comprobando así que es la correcta y también verificando que en los acuerdos (ya sean de licencias cruzadas o de derechos de explotación) toda la información necesaria esté en orden.

### **3.1.5 Territorialidad de las patentes**

Como se ha venido indicando a lo largo del presente Proyecto, hay información relacionada con cada patente que es de especial importancia. Entre los datos más relevantes que se encuentran dentro de cada licencia, podemos destacar el informe del estado de la técnica, la fecha de solicitud y emisión de la patente y, por último, la territorialidad de la patente.

Dentro de los metadatos de cada NFT que representa una patente, irá incluido la territorialidad de la misma, es decir se establecerá dónde se ha presentado y otorgado el derecho de patente.

Uno de los objetivos del uso de la tecnología blockchain, es que el mundo entero esté conectado, significando esto que el sistema de patentes no entienda de barreras geográficas para limitar el derecho de esas patentes. Evidentemente, es un cambio que no se puede realizar de forma repentina, si no que tendrá que realizarse de forma paulatina, estableciendo ese derecho global para las nuevas patentes de ahora en adelante.

## **Cálculo de los precios de las patentes.**

En el presente Modelo, como se ha venido mencionando, se busca optimizar el sistema de patentes actual por medio de la automatización. Entonces, para llevar esto a cabo, se ha elaborado un modelo que calculará los precios de las patentes de forma prácticamente automática.

Una vez el usuario ha obtenido la licencia de la patente y esta se añadido al sistema que utiliza la tecnología blockchain, el programa le presenta una serie de preguntas que sirven para conocer cuál es su posición frente a futuras licencias. Estas preguntas servirán para categorizar las patentes dentro de nuestros parámetros establecidos para así darle el valor que se considere oportuno siguiendo la lógica del modelo. Los aspectos en los que se cataloga la patente son los siguientes:

- Complejidad de la tecnología
- Uso específico
- Tipo de licencia (exclusiva o no exclusiva)
- Duración del acuerdo
- Tipo de industria
- Duración de la licencia

Cada uno de estos parámetros tendrá después un coeficiente asignado que influirá en el precio final de la patente. Todos los coeficientes multiplican una cantidad inicial cuyo valor se ha calculado teniendo en cuenta el coste de la patente a lo largo de los años, tanto el de creación como el de mantenimiento.

En primer lugar, definiremos los distintos coeficientes, explicando cada valor asociado y el razonamiento detrás de cada uno. Después desarrollaremos la cantidad inicial  $c$ , así como los datos que nos han llevado a darle su valor.

Entre los coeficientes encargados de fijar el valor final de la patente, se encuentran los siguientes:

- **t**, hace referencia al tipo de tecnología que conlleva el desarrollo de la patente. El valor de este varía de 0.9 a 1.1 con intervalos de 0.1. Se considera importante reconocer el trabajo que hay detrás del desarrollo de una patente, premiando con más valor a aquellas más complejas.
- **u**, viene a representar si el uso de la patente es específico o no. Una patente que puede ser utilizada en distintos ámbitos aporta un mayor número de posibilidades que aquella con un único uso. Se considera oportuno premiar la versatilidad de una patente.
- **e**, viene a representar la exclusividad de la patente. En caso de que fuera de uso exclusivo, se estaría limitando la posibilidad de aumentar los ingresos del licenciatario frente a otras concesiones de licencias de explotación. Por ello en caso de que una patente fuese de uso exclusivo, el precio final de la licencia aumentaría frente al que tendría si pudiera haber más de un licenciatario.
- **d**, hace referencia a la duración del acuerdo. Si se quisiera obtener una licencia por muchos años, el licenciatario se estaría asegurando el acceso a esa tecnología. Por ellos, se ha considerado oportuno aumentar el valor de la licencia según aumenta la duración de esta.
- **s**, viene a representar la situación del licenciante y el licenciatario en la industria. En caso de que una nueva empresa quisiera entrar a un mercado en el cual no cuenta con experiencia y necesita nuevas licencias de explotación, el valor de la licencia no se vería afectado. Pero, por el contrario, si ambos licenciante y

licenciatario pertenecieran a la misma industria, el valor final de la patente sería menor. Este razonamiento se debe a que, como se ha mencionado anteriormente en el Proyecto, muchas de estas empresas buscan obtener licencias de manera prácticamente continua, están apostando por la innovación y el desarrollo en un mercado altamente competitivo como podría ser el farmacéutico o el de los microprocesadores. Es lógico pensar que, entre empresas del mismo mercado, se ayuden entre ellas para potenciar el desarrollo de ambas. Aunque sea un mercado competitivo, hemos visto antes que de nada serviría copiar una patente en un ambiente de constante evolución y desarrollo porque uno se quedaría atrasado.

- **n**, representa el número de países dónde se está solicitando la licencia frente al número de países totales donde se tiene la misma. Se ha considerado oportuno aumentar el valor final de la patente si un licenciatario tuviera la licencia en todos los países donde existe la patente.

En la siguiente tabla se muestran los distintos valores que pueden tomar los coeficientes en base a las condiciones anteriormente expuestas. En el caso del coeficiente que hace referencia al número de países donde se solicita o la duración del acuerdo, se ha expresado el valor del coeficiente con la función indicada en la tabla. Se considera necesario el uso de tablas en este formato para facilitar así la comprensión del cálculo:

La siguiente Figura 6 muestra los distintos coeficientes, así como sus valores utilizados que multiplican la cantidad inicial **c**:

| Coeficientes                     |   |                      |                 |
|----------------------------------|---|----------------------|-----------------|
| Complejidad de la tecnología (t) | <u>Básica</u>   | <u>Media</u>         | <u>Avanzada</u> |
|                                  | 0.9   | 1                    | 1.1             |
| Uso de la patente (u)            | <u>Específico</u>   | <u>No específico</u> |                 |
|                                  | 0.9   | 1.1                  |                 |
| Duración del acuerdo (d)         | $d = 1 + \frac{\text{número de años}}{20}$  |                      |                 |
| Tipo de licencia (e)             | <u>Exclusiva</u>  | <u>No exclusiva</u>  |                 |
|                                  | 1.1   | 1                    |                 |
| Misma industria (s)              | <u>Si</u>   | <u>No</u>            |                 |
|                                  | 0.8   | 1                    |                 |
| Número de países (n)             | $n = 1 + \frac{\text{número de países seleccionados}}{\text{número de países totales}}$ |                      |                 |

**Figura 6: Tabla de coeficientes. Elaboración Propia. 2023**

La cantidad conocida como inicial, se ha calculado realizando una curva de precios, donde en base a la duración restante de la licencia en posesión del licenciatario, su precio varía. Gracias a esto, se ha conseguido mantenerse objetivo a la hora de valorar las patentes, automatizando y por tanto agilizando el proceso.

Según la OEPM, el coste de obtención de una patente son 1198,49 euros, de los cuales 691,5 euros corresponden a la solicitud del estado de la técnica, y, por otro lado, la petición del examen sustantivo que asciende a 393,67 euros. Esta cantidad, está excluyendo las tasas para mantener la licencia de la patente que en este caso comienzan en 18,66 los primeros tres años y van aumentando de forma progresiva como muestra la siguiente Figura 7:

|     | Clave | €      |
|-----|-------|--------|
| 3ª  | IP03  | 18,66  |
| 4ª  | IP04  | 23,29  |
| 5ª  | IP05  | 44,55  |
| 6ª  | IP06  | 65,75  |
| 7ª  | IP07  | 108,54 |
| 8ª  | IP08  | 135,12 |
| 9ª  | IP09  | 169,56 |
| 10ª | IP10  | 218,22 |
| 11ª | IP11  | 273,53 |
| 12ª | IP12  | 321,16 |
| 13ª | IP13  | 368,70 |
| 14ª | IP14  | 416,69 |
| 15ª | IP15  | 445,00 |
| 16ª | IP16  | 463,44 |
| 17ª | IP17  | 494,90 |
| 18ª | IP18  | 494,90 |
| 19ª | IP19  | 494,90 |
| 20ª | IP20  | 494,90 |

*Figura 7: Precio patentes a lo largo de los años. Elaboración Propia. 2023*

Entonces, hasta ahora, tendríamos contabilizado el coste de las tasas necesarias para tanto la creación de la patente como para mantenerlas. Por último, habría que sumar unos gastos que irían asociados al sistema, es decir un cobro que realizaría el sistema por la gestión de los trámites de la licencia. En este caso, se ha seleccionado la cantidad de 7500 euros, cantidad media aproximada de los datos que se han observado en la mayoría de los casos como asociados a gastos de agentes de la propiedad. Entonces, si sumamos el precio base de las patentes, es decir los gastos de gestión iniciales, y a eso le añadimos el precio a pagar a medida que avanza el tiempo de la licencia y los gastos del agente de la propiedad, obtenemos la siguiente Figura 8:

Precio en euros de una licencia de patente a lo largo de los años

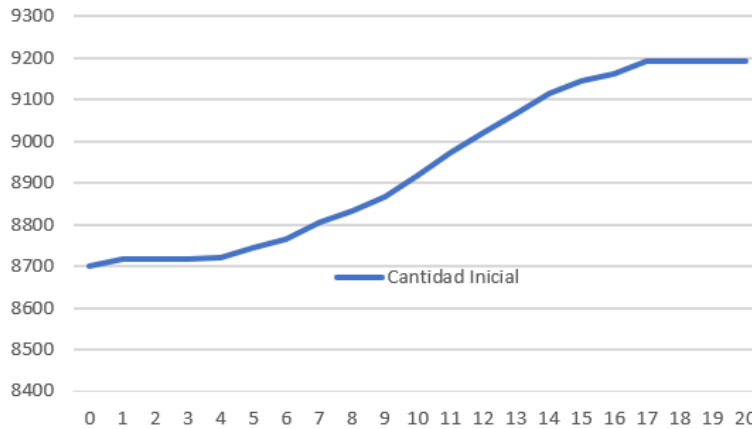


Figura 8: Curva de Precios de una patente a lo largo de los años. Elaboración Propia. 2023

A la hora de evaluar cual sería la cantidad inicial, se sumaría la cantidad necesaria a pagar para mantener la licencia y el coste de la licencia del punto en el que se encuentra. Es decir, si por ejemplo una licencia se fuese a solicitar por cinco años en el año dos de licencia, la patente tendría un valor inicial de  $8717.5 + 170.91$  de tasas de mantenimiento (acumulando las cantidades de los años dos al seis), es decir, un total de 8888.06 euros.

Por último, para completar el ejemplo donde se solicita una licencia durante cinco años empezando en el año dos, faltaría completar las especificaciones que determinan los valores de los coeficientes. Si fijamos, por ejemplo, que la licencia sea exclusiva, el uso de la patente sea específico, licenciante y licenciatarario no trabajen en la misma industria, se esté solicitando en dos de cinco países posibles y la tecnología sea avanzada, el precio final de la licencia se calcularía de la siguiente forma:

$$p = c * e * t * u * d * s * n \quad (1)$$

Asignando los valores de los coeficientes para este caso:

$$p = 8888.06 * 1.1 * 1.1 * 0.9 * 1.05 * 1 * 1.05 \quad (2)$$



$$p = 10671.2 \text{ euros} \quad (3)$$

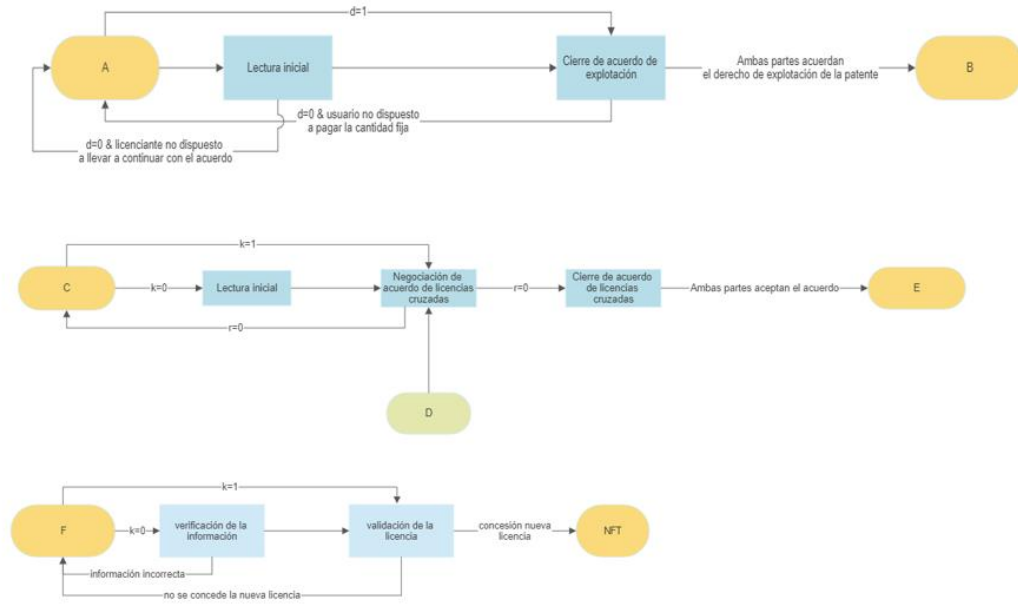
Donde  $p$  representa el precio final de la licencia calculado por nuestro modelo. De esta forma, se estaría calculando el precio de la patente de manera automática una vez se fijarán las condiciones por medio del cuestionario que realizaría el usuario. En el siguiente capítulo, se pondrá en práctica el modelo con distintos casos.

### **3.2 Estructura y funciones del modelo**

En esta parte del capítulo se presentarán y explicarán las estructuras de las principales funciones del modelo. Todas las funciones se explicarán de forma similar, detallando en primer lugar la estructura para después describir las variables y su intención dentro del modelo. Por último, se mostrará el código programado con la herramienta de la inteligencia artificial para automatizar la función.

El modelo en cuestión aplicará las características de la tecnología blockchain a la creación de patentes y los acuerdos de licencias cruzadas. Con esto se busca optimizar el proceso de creación de las patentes desde el punto de vista económico y también respecto al ahorro de tiempo. Además, también se elaborará una estructura la cual sirva para automatizar todos los pagos necesarios en el sistema de patentes actual.

A continuación, en la Figura 9, se presenta la estructura general de nuestro modelo, que será explicada por partes, así como todas sus variables.



**Figura 9: Esquema general de las funciones del modelo. Fuente: Elaboración Propia, 2023**

En la Figura 1, se pueden apreciar tres funciones diferenciadas, cada una con una variable de entrada distinta, tres tensores, cada uno intencionado a una parte distinta de la estructura. La estructura de cada una de las tres funciones se desglosará de manera progresiva para facilitar así la correcta explicación de esta.

Todas las funciones se han programado en Solidity con la ayuda de la inteligencia artificial. Se ha considerado fuera del alcance del proyecto desarrollar los programas en su totalidad debido a su alta complejidad, por lo que, por medio de la herramienta de inteligencia artificial, introduciendo las condiciones del sistema minuciosamente se ha logrado programar correctamente el modelo.

### 3.2.1 Solicitar la explotación de una patente.

En esta función, una vez el usuario se haya registrado en el sistema, será capaz de solicitar el derecho de explotación de una patente de manera descentralizada y sin la necesidad de mantener contacto directo con el licenciante. El usuario en cuestión ingresará en el sistema un tensor **A**, donde:

$$\mathbf{A} = (\mathbf{a}_i, \mathbf{s}_j, k) \quad (4)$$

Las variables, que se explicarán a continuación quedan definidas de la siguiente forma:

- $\mathbf{a}_i$  representa el número de licencia de cada patente.
- $\mathbf{s}_j$  representa el numero identificador de empresa poseedora de la licencia de la patente
- $k$  representa el avance del usuario dentro del proceso del sistema.

La primera y la segunda variable servirán para identificar a la empresa de interés y cuál es la patente que interesa al usuario. La última de las variables del tensor también permitirá al usuario, dependiendo del valor que introduzca, continuar con un proceso ya empezado. Si el usuario le diera el valor 0 a esa variable, estaría empezando un proceso de solicitud del derecho de explotación de una patente. Por otro lado, si introdujera el 1, el sistema interpretaría que para ese número de patente y para esa empresa ya existe un proceso comenzado y le llevaría a él.

Una vez se ha interpretado la variable de entrada, el siguiente paso sería verificar que el licenciante estaría dispuesto a negociar un acuerdo de explotación de una patente. Previamente, el licenciante, a la hora de subir toda la información referente a una patente, habría decidido si estuviera dispuesto a recibir ofertas por su patente. Respondiendo a las preguntas del sistema, encargadas de categorizar la patente dentro de los parámetros establecidos para el posterior cálculo de su precio. En caso de no estar dispuesto, al usuario que hubiera ingresado ese número de patente, se le informaría de que no está disponible y se le llevaría de nuevo al inicio del modelo.

En caso de que sí aceptara ofertas, al usuario se le mostrarían los datos de contacto del licenciante junto con la cantidad que solicita para cerrar el acuerdo de explotación de una patente de manera directa. Si el usuario en cuestión no quisiera negociar y por tanto no mantener contacto, podría pagar esa cantidad establecida por el licenciante para cerrar el acuerdo.

Tras realizar el pago al licenciante con la criptomoneda ether por medio de la blockchain de Ethereum, se estaría realizando también una llamada al contrato

inteligente que habilitaría al usuario en cuestión el acceso a los metadatos del NFT que contienen toda la información que protege la patente. Además, el sistema emitiría un tensor llamado **B**, donde **B**:

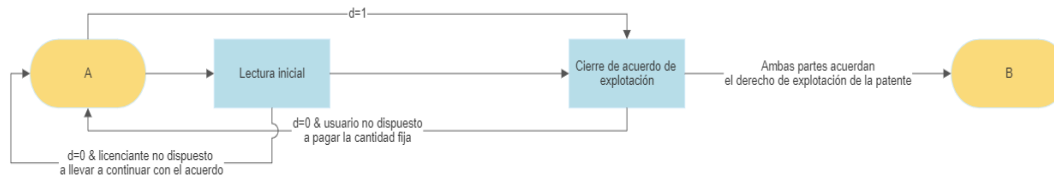
$$\mathbf{B} = ( \mathbf{a}_i , \mathbf{s}_j , \mathbf{p}_t , \mathbf{p}_r , \mathbf{h} ) \quad (5)$$

Las variables del tensor en cuestión serían las siguiente:

- $\mathbf{a}_i$  representa el número de licencia de cada patente.
- $\mathbf{s}_j$  representa el numero identificadorio de empresa poseedora de la licencia de la patente.
- $\mathbf{p}_t$  representa la fecha de iniciación de la validez del acuerdo de explotación de la licencia.
- $\mathbf{p}_r$  representa la fecha de finalización de la validez del acuerdo de explotación de la licencia.
- $\mathbf{h}_i$  representa el número asociado a la transacción que conlleva el acuerdo en la cadena de bloques de Ethereum.

Este tensor de salida compartirá las primeras dos variables con el tensor **A**, además incluirá entre sus componentes las fechas de iniciación y finalización del acuerdo de explotación. Por último, la última variable consistirá en el hash de la transacción que se genera de forma automática al ejecutar el contrato inteligente en la red de Ethereum. De esta forma, se podrá localizar fácilmente el acuerdo en la cadena de bloques quedando este registrado para siempre y a la vista de todo el mundo.

A continuación, se muestra en la Figura 10 un esquema de la actual función descrita:



**Figura 10: Esquema de la primera función: Solicitud de explotación de una patente. Fuente: Elaboración Propia, 2023**

Las condiciones del acuerdo estarían previamente fijadas por el licenciante e incluidas en el smart contract que lo gestiona. Si se diera el caso en el que se cumpliera alguna de las condiciones y el licenciataria tuviera que pagar un royalty al licenciante, este pago se realizaría de forma automática y prácticamente instantánea por medio de la red de Ethereum.

De esta forma se habría realizado un acuerdo para la explotación de una patente de manera descentralizada y segura, todo ello por medio de la tecnología blockchain. Además, toda la información del proceso estaría recogida dentro de la cadena de bloques donde quedarían anotadas las condiciones, la fecha de inicio del acuerdo y la fecha de finalización de este.

El código de esta función, que se puede consultar en el Anexo A, es un contrato inteligente programado en Solidity que permite gestionar la solicitud y adquisición de patentes mediante la utilización de criptomonedas Ether. Por medio de la inteligencia artificial, detallando minuciosamente el funcionamiento del programa se ha llevado a cabo la programación del código.

### 3.2.2 Acuerdo de licencias cruzadas

Esta función del modelo permite al usuario proponer un acuerdo de licencias cruzadas, para más tarde en caso de que ambas partes acepten, llevarlo a cabo. Ambas empresas necesitan las licencias de patentes de la otra para poder comercializar productos sin riesgo de incumplir la explotación de patentes. En este caso la variable de entrada es un tensor  $C$ , donde:

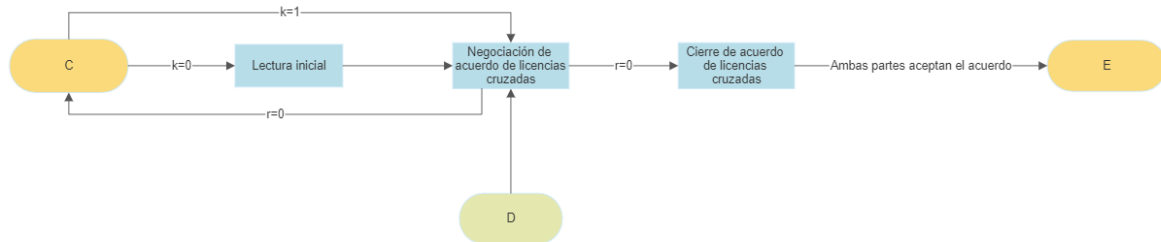
$$C = ( \mathbf{q}_i, \mathbf{s}_i, \mathbf{s}_j, \mathbf{c}_i, k ) \quad (6)$$

Las variables del tensor  $C$  quedan definidas de la siguiente forma:

- $\mathbf{q}_i$  representa un vector cuyas componentes son el número de licencias de las patentes de interés.
- $\mathbf{s}_i$  representa el número identificador de la empresa poseedora de las patentes de interés.
- $\mathbf{s}_j$  representa el número identificador de la empresa interesada en el acuerdo.
- $\mathbf{c}_i$  representa el texto donde irán incluidas las condiciones propuestas por la empresa  $\mathbf{s}_i$ .
- $k$  es una variable booleana que representa el avance del usuario (empresa  $\mathbf{s}_i$ ) dentro del proceso del sistema.

En caso de que el usuario hubiera empezado el proceso previamente este podría asignar el valor 1 a  $k$  (variable booleana) indicando que ya existe un progreso previo y el sistema le redirigiría directamente ahí.

El esquema de la actual función, que será explicado a continuación, es el siguiente de la Figura 11:



**Figura 11: Esquema de la segunda función: Acuerdo de licencias cruzadas. Fuente: Elaboración Propia, 2023**

Una vez se ha verificado toda la información de entrada, al usuario se le pondría en contacto con la empresa de interés para que ambas partes lleguen a un acuerdo. En caso de llegar a ese acuerdo, ambas empresas establecerían las bases de este, incluyendo en ellas las patentes abarcadas en el acuerdo de cada una, así como las distintas cláusulas y los royalties que habría que pagar. En caso de que no sea un intercambio que cuente con la misma cantidad de licencia cedidas de patentes por cada una de las partes y se incluya una cierta cantidad económica, el pago de esta se realizaría por medio de las criptomonedas a través de la red de Ethereum.

La segunda empresa implicada, una vez ha contactado con la primera y ha aceptado negociar, entrará en el sistema con el tensor **D**, donde:

$$D = ( \mathbf{q}_j , \mathbf{s}_j , \mathbf{c}_j , \mathbf{r}_j ) \quad (7)$$

Las variables del tensor tienen el siguiente significado:

- $\mathbf{q}_j$  es un vector cuyas componentes son los números de licencias de las patentes de interés para la empresa  $s_j$
- $\mathbf{s}_j$  representa el número identificador de la empresa que no inicia el acuerdo.
- $\mathbf{c}_j$  representa el archivo de texto donde irán incluidas las condiciones propuestas por la empresa  $s_j$
- $\mathbf{r}_j$  representa la intención de la empresa  $s_j$  frente a la negociación.

Por medio de este tensor la empresa encargada de recibir la propuesta de acuerdo sería capaz de establecer sus condiciones en la negociación, así como las patentes que son de su interés. También por medio de la variable booleana  $\mathbf{r}_j$ , si la empresa  $s_j$  le diera el valor

0, la negociación no se llevaría a cabo, si por el contrario el valor de esta fuese 1, se procedería con la negociación.

Toda la información del acuerdo iría incluida dentro de un archivo de texto llamado “c”, donde cada parte establecería sus bases para el acuerdo, así como la información relevante de cada patente de cara al acuerdo. Por último, se realizaría la llamada al smart contract, para así llevar a cabo el acuerdo, y que este quedase registrado en la cadena de bloques.

Tras realizar la llamada al contrato inteligente y haber ejecutado la transacción, el smart contract acuñaría un NFT para cada dirección en la cadena de bloques de cada empresa que le serviría como certificado digital del acuerdo concediéndole el permiso de explotación de la patente, así como toda la información que protege la misma.

La variable de salida de esta función del sistema sería un tensor **E**, donde:

$$E = ( \mathbf{q}_{ij} , s_i , s_j , \mathbf{p}_r , \mathbf{p}_t , \mathbf{c}_{ij} , \mathbf{h}_{ij} ) \quad (8)$$

Las variables anteriormente mencionadas quedarían definidas:

- $\mathbf{q}_{ij}$  vector que contiene los distintos números de las patentes implicadas en el acuerdo.
- $s_j$  representa el número identificador de la empresa que inicia el acuerdo.
- $s_i$  representa el número identificador de la empresa que recibe la propuesta.
- $\mathbf{p}_r$  representa la fecha de inicio de la validez del acuerdo de explotación de las patentes contenidas en  $\mathbf{q}_{ij}$  donde participan las empresas  $s_i$  y  $s_j$ .
- $\mathbf{p}_t$  representa la fecha de finalización de la validez del acuerdo de explotación de las patentes contenidas en  $\mathbf{q}_{ij}$  donde participan las empresas  $s_i$  y  $s_j$ .
- $\mathbf{c}_{ij}$  representa el texto donde irán incluidas las condiciones propuestas por ambas empresas.
- $\mathbf{h}_{ij}$  representa el hash asociado a la transacción del acuerdo en la blockchain.

El tener en cuestión contaría con toda la información referente al acuerdo entre sus variables: las empresas implicadas, las patentes de interés para ambas, el tiempo por el que es válido, las condiciones impuestas por cada una de las dos partes y, por último, el



hash de la transacción que emite el contrato inteligente al ejecutarse. De esta forma el contrato quedaría registrado en la blockchain, siendo inmutable frente a modificaciones.

Todas las condiciones del acuerdo estarían previamente fijadas y acordadas por ambas partes. En caso de que alguna empresa implicada incumpliera alguna de las cláusulas, el smart contract emitiría una transacción donde negaría el acceso a una de las dos partes.

Por lo tanto, con esta función se estaría llevando a cabo la labor que realizan actualmente las oficinas de patentes, pero sin la necesidad de depositar la confianza de uno en un intermediario. Todo el proceso de un acuerdo de licencias cruzadas entre empresas se estaría gestionando por medio de la tecnología blockchain y los smart contracts, donde se habría prácticamente automatizado todo por medio de código.

El código que define el programa para llevar a cabo esta función se puede encontrar en el Anexo B. Como se ha comentado anteriormente, la programación en su totalidad de la función se escapa de las capacidades de este proyecto por lo que se ha recurrido a una herramienta como es la inteligencia artificial.

### **3.2.3 Creación de una nueva patente**

Por último, la tercera función en cuestión del modelo sería la de creación de una patente, por medio de esta el usuario podrá realizar todos los trámites de manera descentralizada y segura por medio de la tecnología blockchain, ahorrando así en tiempo y dinero.

El esquema de funcionamiento en este caso es el más simple de todos los planteados. En primer lugar, al sistema entrará un tensor  $\mathbf{F}$ , donde:

$$\mathbf{F} = (t, \mathbf{p}_t, k) \quad (9)$$

Las variables del tensor anteriormente mencionada quedarían descritas de la siguiente forma:

- $t$  representa un archivo de texto donde irá incluida toda la información relevante de la licencia que se busca obtener.
- $\mathbf{p}_t$  representa la fecha en la que se crea la solicitud de la patente

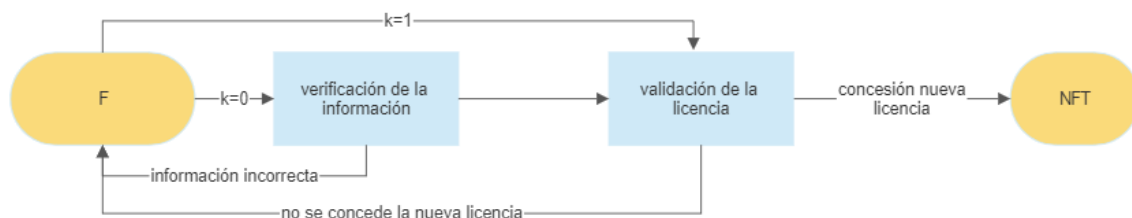
- $k$  es una variable booleana utilizada para determinar en qué parte del proceso se encuentra el usuario.

Este tensor contará entre sus variables con la información del nuevo método, proceso o producto que se quiere patentar junto con el informe de estado de la técnica, todo ello incluido en un archivo de texto, también, por otro lado, tendremos como variables del tensor la fecha en la que se crea la solicitud de esta nueva patente y una variable booleana donde se indica si el proceso ha sido empezado previamente. Esto último servirá como ayuda para posibles conflictos que pueda aparecer donde existan procesos de fabricación que utilicen un proceso ya patentado.

Al incluir el informe del estado de la técnica, se estarán localizando procedimientos similares al propuesto para patentar en cualquier país sobre una determinada área tecnológica, probando así que el método/producto es necesario junto con problemas técnicos que viene a solucionar. También, en el mismo se incluir los números de referencias de las otras patentes que son utilizadas en la elaboración del producto.

Toda esta cantidad de información subida a la red permitirá al usuario buscar lo que necesite para elaborar una nueva solicitud de patentes, al estar sobre la blockchain, no entiende de barreras territoriales ni de horarios, por lo que las gestiones de búsqueda se podrían llevar a cabo desde cualquier punto del mundo a cualquier hora, sin depender así de una entidad central encargada de suministrar la información.

El esquema en cuestión es el siguiente en la Figura 12:



**Figura 12: Esquema de la tercera función: Creación de nueva patente. Fuente: Elaboración Propia, 2023**

Una vez el usuario ha interactuado con la dApp del modelo planteado y ha subido toda la información necesaria para comenzar el proceso, comienza la fase de verificación por parte de aquellos nodos que tienen la potestad para ello. Como se ha

explicado previamente, estos nodos no son los encargados de validar las transacciones en la red de Ethereum, sino que son los responsables de juzgar que toda la información subida a la cadena de bloques por parte del usuario es la correcta y cumple todos los requisitos necesarios.

No es necesario que el usuario suba toda su invención directamente, podrá realizarlo por medio de transacciones en oculto, de tal forma que, una vez se haya validado su solicitud y la transacción de validación del comienzo de solicitud haya tenido lugar, parte de información será revelada, para evitar así posibles apropiaciones de ideas por parte de los validadores.

Estos validadores del sistema, serán atribuidos con ese título por medio de la oficina de patentes, que, tras verificar que los poseedores de esa dirección en la cadena de bloques son personal cualificado para realizar la función de validar la información subida, se les enviará por medio de la blockchain un activo criptográfico digital (NFT) el cual les concederá el acceso de validadores a la cadena de bloques.

Una vez se valide que toda la información es correcta, al usuario en cuestión que haya solicitado la creación de esa nueva patente, el smart contract encargado de gestionar esta función emitirá un NFT donde se acreditará la posesión de esa licencia de patente, así como toda la información que contiene la patente. De ahora en adelante si algún usuario quisiera acceder a la información de la patente tendría que utilizar la función del actual sistema planteado encargada de ello.

Por último, después de que la oficina encargada de gestionar la licencia de la patente haya introducido la licencia al sistema que interactúa con la tecnología blockchain, el licenciatarario deberá responder a unas preguntas para conocer su posición frente a futuras licencias. De esta forma, por medio de estas preguntas y en base a las respuestas dadas, el modelo calcularía el precio para futuras licencias de manera automática.

El código que define el programa planteado se ha elaborado con ayuda de la herramienta de la inteligencia artificial detallando las características de la función. El código en cuestión se puede consultar en el Anexo C.

### 3.3 Conclusiones

Para concluir, tras explicar cada una de las distintas funciones del modelo, se puede apreciar que gran parte de los procesos se realizarán de forma automatizada. Cada una de las funciones, como hemos visto, se basa en un tensor de entrada que interactúa con un contrato inteligente, para después devolver un tensor de salida o un NFT como es en el caso de la creación de patentes.

También, otra parte que impulsa la automatización en el presente Modelo es el cálculo automatizado del precio de las licencias. Por medio de un simple cuestionario, se ha tratado de actuar de forma imparcial y justa para así ahorrar tiempo y dinero en las negociaciones.

En su totalidad el modelo depende de muchos factores distintos, pero en el presente capítulo se ha tratado de concretarlos al máximo, por un lado, basándonos en las ideas recopiladas en la parte informativa del Proyecto y por otro con ideas propias que se ajustan a nuestros intereses en el Proyecto.

En primer lugar, la función para solicitar una licencia de explotación recibe el tensor **A**, este, como hemos visto cuenta entre sus variables con toda la información relevante al acuerdo. En caso de que la empresa licenciante aceptara el acuerdo, todas las condiciones se incluirían dentro del contrato inteligente, eliminando así la necesidad de intermediarios que validasen el acuerdo, llevando todo a cabo de forma descentralizada y autónoma

En segundo lugar, la función para solicitar un acuerdo de licencia cruzadas también recibe un tensor, en este caso **C**, que también comparte algunas variables con **A**. En esta función, también introduce un tensor la segunda empresa implicada en el acuerdo, en este caso entra al sistema con el tensor **B**. Por último, estos dos llegarían a un acuerdo y dejarían todas las cláusulas de este incluidas dentro del archivo de texto  $c_{ij}$ .

En tercer lugar, la función para crear una patente requiere que el usuario introduzca un tensor **F** con la información que sería necesaria para validar la licencia en el método tradicional, pero, en este caso no se devolvería un tensor de salida, si no un NFT cuyos metadatos incluyen toda la información relativa a la licencia de la patente.

A esta automatización se le podrá atribuir un ahorro en tiempo y dinero, para conocer la cantidad cuantitativa con cierta exactitud será necesario comparar con casos prácticos reales. Esto se llevará a cabo en el siguiente capítulo donde se analizarán casos de forma paralela con el modelo propuesto y con el sistema de patentes actual.

## 4. Aplicaciones prácticas del Modelo

En el presente capítulo se pondrán en práctica las funciones del modelo propuesto en el capítulo previo. Para llevar a cabo esto, se plantearán casos reales adaptados al Modelo aplicando las características de la tecnología blockchain y los smart contracts.

Se realizarán casos prácticos para cada función del Modelo para tratar así de analizar el modelo en su totalidad. Comenzaremos con los casos de solicitud de una patente, continuaremos con un ejemplo de cómo sería el proceso de solicitud de una licencia de explotación de una patente y acabaremos con los casos prácticos de la función encargada de gestionar acuerdos de licencias cruzadas. Durante el desarrollo de los casos prácticos, se irán mostrando en paralelo imágenes de la interfaz que se le presentan al usuario al utilizar el sistema.

Por último, terminaremos el capítulo con unas breves conclusiones de la puesta en práctica de cada función ya que en el próximo capítulo estas se realizarán más a fondo y desde distintos enfoques.

### 4.1 Explicación previa de los valores establecidos de inicio.

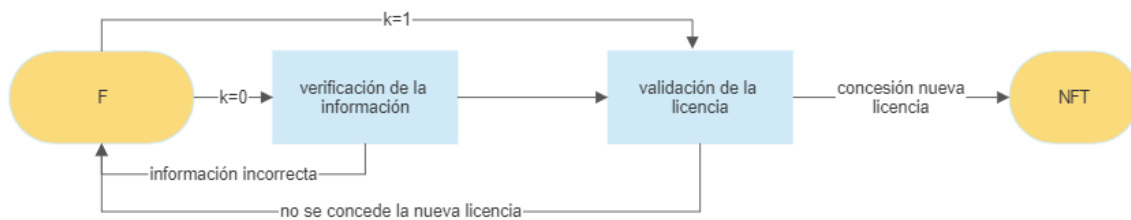
Existen algunos parámetros dentro del Modelo que requerirían de la puesta en práctica de este para poder obtenerlos, por ello, estos valores se asignarán de manera aleatoria, especificando su valor previamente. Entre estos valores o datos se encuentran los números identificatorios de las empresas, las condiciones acordadas, la dirección en la blockchain para cada empresa o los hashes de las transacciones al ejecutar el acuerdo.

Además, habrá valores que el sistema recogerá, y que son necesarios para su funcionamiento, como la dirección en la cadena de bloques o el número identificatorio de la empresa, tanto licenciante como licenciataria, pero que irán asociados a otros valores como el número de patente o con el inicio de sesión en el sistema.

## 4.2 Caso Practico de solicitud de la licencia de una nueva patente

En el actual subcapítulo desarrollaremos un caso práctico utilizando la función encargada de gestionar una creación de una nueva patente por medio de la tecnología blockchain y los contratos inteligentes. Para llevar a cabo el caso práctico, en primer lugar, tendremos que adaptar la información de entrada necesaria a nuestro proyecto. Además, también tendremos que establecer aquellos valores cuya obtención de estos se escapa del alcance del Proyecto.

De nuevo, se considera importante recordar la estructura de la función para facilitar así la comprensión del caso práctico. En la Figura 13 (hay que cambiarlo) que se muestra a continuación se observa el esquema de flujos de la función.



**Figura 33: Esquema de la primera función: Solicitud de explotación de una patente. Fuente: Elaboración Propia, 2023**

La figura anterior muestra el esquema general de la función donde entra un tensor  $F$ , cuyas variables almacenan la información necesaria a la hora de solicitar una patente. Después, aquellos usuarios denominados como validadores en el Modelo se encargarían de verificar y validar toda la información. Por último, en caso de cumplir todos los requerimientos, al solicitante se le concedería la licencia de la nueva patente en forma de NFT, cuyos metadatos contendrán toda la información de la licencia y del proceso o producto que protege la misma. Además, para esta parte práctica se añadirá la parte que incluye la gestión de la patente frente a futuras licencias por parte de su dueño.

Durante el caso práctico se desarrollará el proceso de obtención de una licencia de patente como si se hubiera llevado a cabo por medio del Modelo. Para este caso se ha escogido la patente de “Artículo absorbente con núcleo absorbente plegado de capa

múltiple” cuyo actual propietario es Attends Healthcare Products Inc, a esta empresa se le ha asignado el número identificador 659854756 (s<sub>5</sub>).

En la Figura 14 (habrá que cambiarlo) se puede apreciar parte de la información relevante referente a esta patente:

|  |  |
|--|--|
|  <p>OFICINA ESPAÑOLA DE<br/>PATENTES Y MARCAS<br/>ESPAÑA</p>  |    |
|  | <p>①) Número de publicación: <b>2 819 181</b></p>  |
|  | <p>⑤) Int. Cl.:<br/><b>A61F 13/15</b> (2006.01)<br/><b>A61F 13/534</b> (2006.01)<br/><b>A61F 13/535</b> (2006.01)</p>  |
| <p>⑫) TRADUCCIÓN DE PATENTE EUROPEA</p>  | <p>T3</p>  |
| <p>⑧6) Fecha de presentación y número de la solicitud internacional: <b>27.02.2015 PCT/US2015/018198</b><br/>         ⑧7) Fecha y número de publicación internacional: <b>03.09.2015 WO15131161</b><br/>         ⑧6) Fecha de presentación y número de la solicitud europea: <b>27.02.2015 E 15754606 (0)</b><br/>         ⑧7) Fecha y número de publicación de la concesión europea: <b>15.07.2020 EP 3110381</b></p> |  |
| <p>⑤4) Título: <b>Artículo absorbente con núcleo absorbente plegado de capa múltiple</b></p>   |  |
| <p>③0) Prioridad:<br/><b>28.02.2014 US 201461946595 P</b><br/><b>06.03.2014 US 201461948744 P</b></p> <p>④5) Fecha de publicación y mención en BOPI de la traducción de la patente:<br/><b>15.04.2021</b></p>  | <p>⑦3) Titular/es:<br/><b>ATTENDS HEALTHCARE PRODUCTS, INC.</b><br/><b>(100.0%)</b><br/><b>1029 Old Creek Road</b><br/><b>Greenville, NC 27834, US</b></p> <p>⑦2) Inventor/es:<br/><b>CHMIELEWSKI, HARRY;</b><br/><b>DUCKER, PAUL;</b><br/><b>VROOMAN, JACOB;</b><br/><b>KAISER, THOMAS A. y</b><br/><b>SERGEANT, TIM</b></p> <p>⑦4) Agente/Representante:<br/><b>GONZÁLEZ PECES, Gustavo Adolfo</b></p> |

*Figura 14: Información relevante de " Artículo absorbente con núcleo absorbente plegado de capa múltiple". OEPM, ES2819181T3 (2020).*

A continuación, se procede a adaptar la información referente a la patente a nuestro modelo por medio de las variables del tensor  $F$ , donde sus variables tienen los siguientes valores:

- $t$  representa un archivo de texto donde irán incluida toda la información de la patente, que será explicada más adelante.
- $p_5$  que representa la fecha en la que se inicia la solicitud, en este caso 20150227.



- $k$ , donde consideraremos que el valor es 0, indicando así que el proceso no se ha empezado anteriormente para analizar el proceso en su totalidad.

Dentro de  $t$  estaría almacenada toda la información referente a la licencia de patente que se quiere obtener, en este caso:

- Título de la patente: “Artículo absorbente con núcleo absorbente plegado de capa múltiple”
- Nombre de los inventores: CHMIELEWSKI, HARRY; DUCKER, PAUL; VROOMAN, JACOB; KAISER, THOMAS A. y SERGEANT, TIM
- Resumen de la patente: *“Un núcleo (200, 400, 500, 600) absorbente que comprende un laminado (100) absorbente de capa múltiple plegado longitudinalmente de al menos tres capas, comprendiendo el laminado (100) absorbente una capa (102) de laminado superior; una capa (104) de laminado inferior; y en el que una capa (106) absorbente está situada entre la capa (102) de laminado superior y la capa (104) de laminado inferior, la capa (106) absorbente comprende más del 90 por ciento en peso de polímero superabsorbente, caracterizado porque - un canal se extiende longitudinalmente a lo largo del laminado absorbente plegado; - un primer conjunto de capas de laminado está situado en un lado del canal y un segundo conjunto de capas de laminado está situado en el otro lado del canal; - una pluralidad de pasajes para líquido situada entre las capas de laminado, - algunos de los pasajes de líquido se abren hacia el canal central y algunos otros pasajes de líquido se abren hacia los lados del laminado plegado, y - bien sea los pasajes que se abren hacia el canal o los pasajes que se abren hacia los lados laminados están separados lateralmente uno de otro para formar un perfil en terraza.”.*
- Descripción detallada de la invención.
- Patentes citadas: Se pueden apreciar en la siguiente Figura 15

|                 |            |            |                                  |   |
|-----------------|------------|------------|----------------------------------|---|
| FR2045623A1 *   | 1969-06-10 | 1971-03-05 | Ruby Ets                         |   |
| US3865112A *    | 1974-03-05 | 1975-02-11 | Kimberly Clark Co                | Small size sanitary napkins with improved absorption capability   |
| US4576596A *    | 1983-03-03 | 1986-03-18 | Kimberly-Clark Corporation       | Resilient shape-retaining sanitary napkin   |
| FR2554390B1 *   | 1983-11-08 | 1986-07-18 | Beghin Say Sa                    | Article a jeter absorbant les liquides  |
| BR9508657A *    | 1994-08-31 | 1997-08-12 | Kimberly Clark Co                | Artigo absorvente   |
| US5803920A *    | 1995-06-07 | 1998-09-08 | Kimberly-Clark Worldwide, Inc.   | Thin absorbent article  |
| JP3675601B2 *   | 1996-09-12 | 2005-07-27 | ユニ・チャーム株式会社                      | 吸収体   |
| US5968027A *    | 1997-03-31 | 1999-10-19 | Mcneil-Ppc, Inc.                 | Absorbent article with coiled side walls  |
| US6632209B1 *   | 1998-03-30 | 2003-10-14 | Paragon Trade Brands, Inc.       | Thin absorbent core made from folded absorbent laminate   |
| US6068620A      | 1998-03-30 | 2000-05-30 | Paragon Trade Brands             | Absorbent laminate  |
| US6124391A *    | 1998-08-18 | 2000-09-26 | Stockhausen GmbH & Co. Kg        | Superabsorbent polymers having anti-caking characteristics  |
| US6794557B1 *   | 1999-07-16 | 2004-09-21 | Associated Hygienic Products Llc | Disposable absorbent article employing an absorbent composite and method of making the same                             |
| JP2001190597A * | 2000-01-06 | 2001-07-17 | Uni Charm Corp                   | 水解性の吸収性物品   |
| BR0002299C1 *   | 2000-05-05 | 2002-11-26 | Johnson & Johnson Ind Com        | Absorvente íntimo, núcleo absorvente e método de fabricação de núcleo absorvente  |
| US6753455B2     | 2001-10-26 | 2004-06-22 | Paragon Trade Brands, Inc.       | Absorbent core attachment   |
| US20030105442A1 | 2001-11-30 | 2003-06-05 | Johnston Lee W.                  | Convection of absorbent cores providing enhanced thermal transmittance  |
| MXPA04005951A * | 2001-12-21 | 2004-11-01 | Procter & Gamble                 | Artículo absorbente desechable que tiene un doblez de barrera para pierna y un doblez externo con elastico para pierna. |

**Figura 15: Patentes Citadas en ES2819181T3. OEPM, ES2819181T3 (2020)**

Una vez el usuario ha introducido toda la información al sistema, los encargados de conceder la licencia de patente son aquellos denominados como validadores. Estos son los responsables de verificar que toda la información es correcta. El usuario que inicio la solicitud puede ir comprobando el progreso de esta, ingresando en el sistema todas las veces que él quiera.

A continuación, se mostrarán imágenes de la parte del proceso que consiste en establecer las condiciones de funcionamiento de la patente frente a futuras licencias por parte de la empresa que ha obtenido la patente.

En primer lugar, en la Figura 16, una vez se ha verificado toda la información, la entidad encargada de conceder la licencia sería la encargada de subir la información de la patente e introducirlo en el sistema que interactúa con la cadena de bloques.



*Figura 16: Interfaz función 3. Elaboración Propia. 2023*

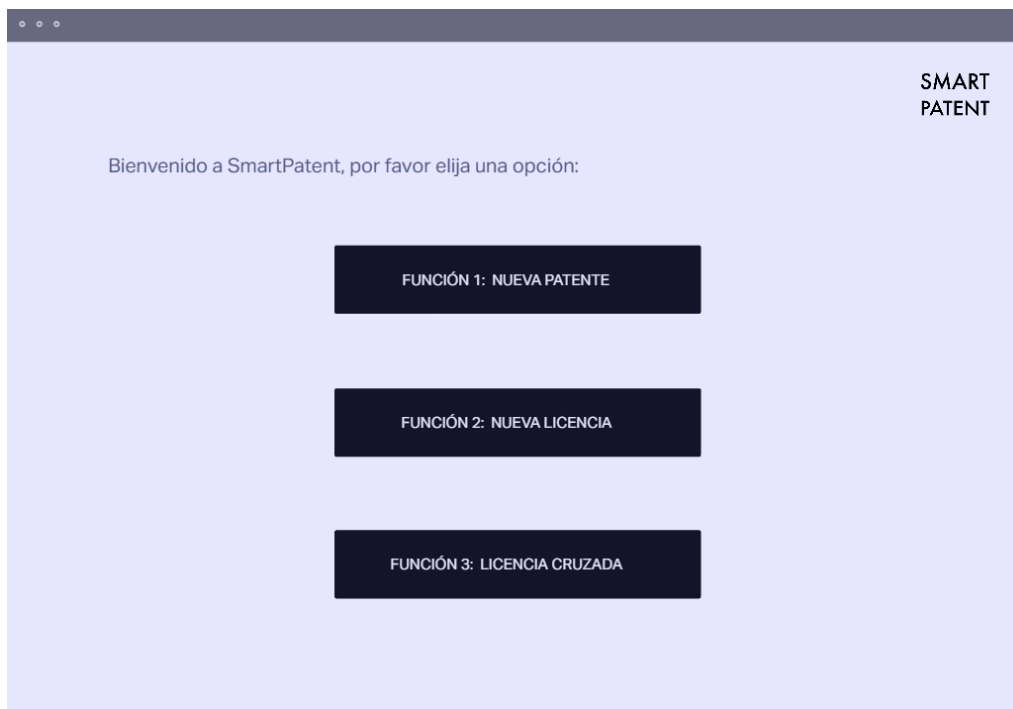
De esta forma, una vez se ha subido la información de la patente y asignado un número identificador a la misma, a la OEPM (en este caso) se le mostraría la siguiente Figura 17, donde se estaría creando un timestamp de la concesión de la patente, quedando todo registrado en la cadena de bloques.



*Figura 18: Interfaz función 3. Elaboración Propia. 2023*

En la anterior Figura 18, se aprecia el mensaje de confirmación que le saltaría a la entidad encargada de validar la información y conceder la patente. En esta función esta sería su última intervención, las siguientes capturas de pantalla vendrían a enseñar la parte del proceso que incumbe a la empresa solicitante.

Entonces, la Figura 19, nos presenta el menú inicial con el que se enfrentaría el usuario. En este caso, como es el objetivo del subcapítulo, se elegirá la Función 1, “Nueva patente”. Una vez se le ha concedido la patente, el usuario recibiría el mensaje que aparece en la Figura 20, especificando la patente que se le ha concedido, cuándo y el número identificador de esta. Además, en esta misma pantalla podría acceder a un mapa con el listado de países donde se le ha concedido la patente.



*Figura 19: Interfaz función 3. Elaboración Propia. 2023*

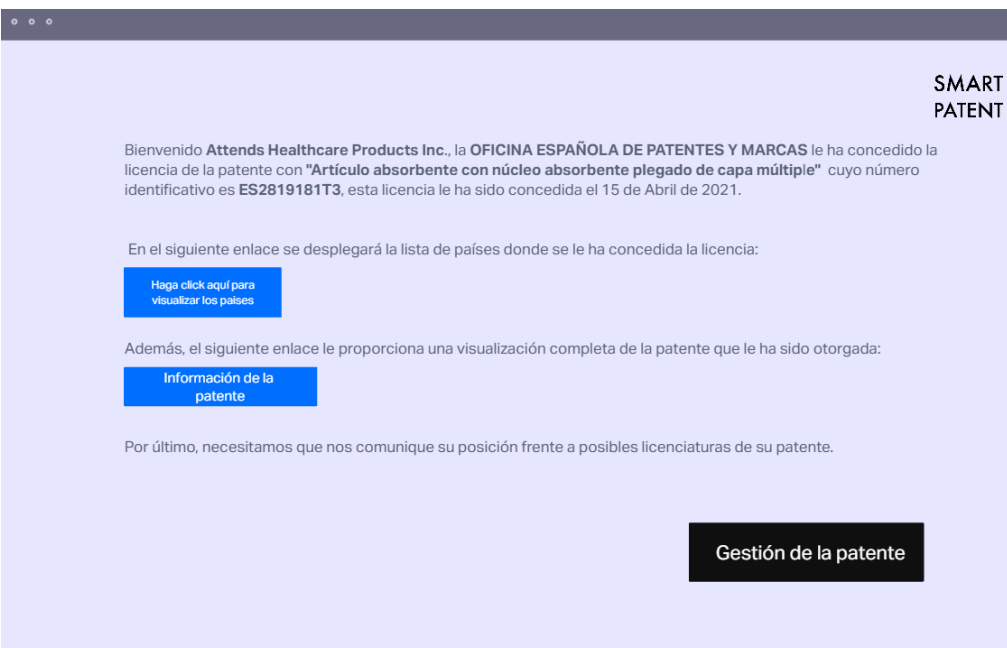


Figura 20 : Interfaz función 3. Elaboración Propia. 2023

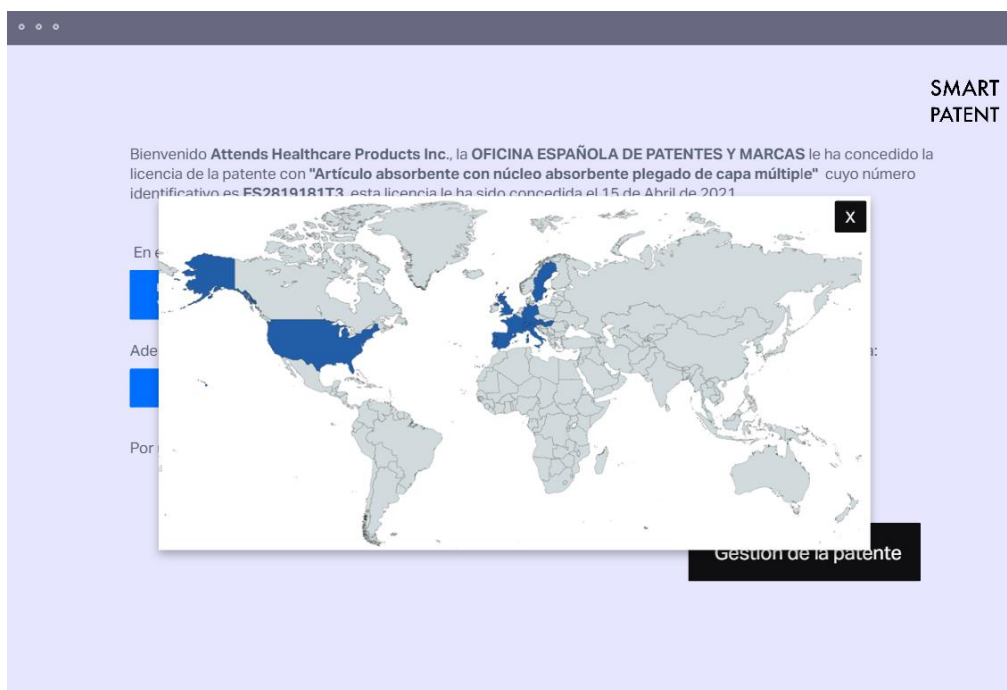


Figura 21: Interfaz función 3. Elaboración Propia. 2023

De esta forma, la anterior Figura 21, muestra los países dónde se tiene la licencia de esa patente. Es importante hacer un pequeño comentario en esta parte relacionado con la obtención de la licencia en los distintos países. Una vez se le concede a la empresa la

patente, esta tiene un año para patentarla en otros países, por lo que, en este caso, la OEPM puede estar concediéndosela ahora debido a que los trámites en España sean más tardíos que en Francia, por ejemplo.

Después, en la siguiente Figura 22, se nos muestra la captura de pantalla que contiene toda la información referente a la patente y que el usuario podría consultar en cualquier momento.



**SMART PATENT**

Número de licencia: ES2819181T3

Empresa Licenciataria: Attends Healthcare Products Inc.

Número identificador de la empresa: 659854756

Título de la patente: Artículo absorbente con núcleo absorbente plegado de capa múltiple

**Resumen:** Un núcleo (200, 400, 500, 600) absorbente que comprende un laminado (100) absorbente de capa múltiple plegado longitudinalmente de al menos tres capas, comprendiendo el laminado (100) absorbente una capa (102) de laminado superior; una capa (104) de laminado inferior; y en el que una capa (106) absorbente está situada entre la capa (102) de laminado superior y la capa (104) de laminado inferior, la capa (106) absorbente comprende más del 90 por ciento en peso de polímero superabsorbente, caracterizado porque - un canal se extiende longitudinalmente a lo largo del laminado absorbente plegado; - un primer conjunto de capas de laminado está situado en un lado del canal y un segundo conjunto de capas de laminado está situado en el otro lado del canal; - una pluralidad de pasajes para líquido situada entre las capas de laminado, - algunos de los pasajes de líquido se abren hacia el canal central y algunos otros pasajes de líquido se abren hacia los lados del laminado plegado, y - bien sea los pasajes que se abren hacia el canal o los pasajes que se abren hacia los lados laminados están separados lateralmente uno de otro para formar un perfil en terraza.

Descargar PDF

Encontrar tecnología anterior

Informe del Estado de la Técnica

Previsualizar PDF

FAMILIA DE PATENTES

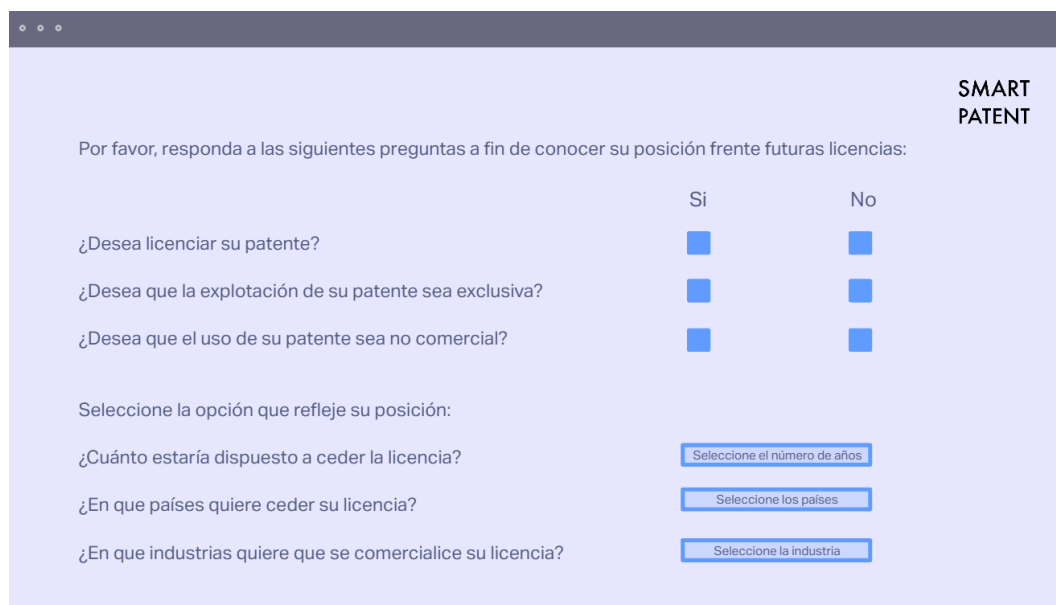
Eventos relacionados con la solicitud

- 2015-02-27 Solicitud presentada por Attends Healthcare Products Inc
- 2021-04-15 Aplicación concedida
- 2021-04-15 Publicación de ES2819181T3

Volver al inicio

Figura 22: Interfaz función 3. Elaboración Propia. 2023

En las últimas capturas, el usuario deberá establecer cuál es su posición frente a futuras licencias. Toda esta información servirá para que después el proceso de solicitud de una licencia de explotación este prácticamente automatizado. En base a las condiciones que la empresa licenciante establezca, el precio de la patente será uno u otro. En el anterior capítulo se explicó el razonamiento detrás de cada coeficiente, así como las cantidades iniciales sobre las que se basa el cálculo.



SMART PATENT

Por favor, responda a las siguientes preguntas a fin de conocer su posición frente futuras licencias:

|  | Si                       | No                       |
|--|--------------------------|--------------------------|
| ¿Desea licenciar su patente?                           | <input type="checkbox"/> | <input type="checkbox"/> |
| ¿Desea que la explotación de su patente sea exclusiva? | <input type="checkbox"/> | <input type="checkbox"/> |
| ¿Desea que el uso de su patente sea no comercial?      | <input type="checkbox"/> | <input type="checkbox"/> |

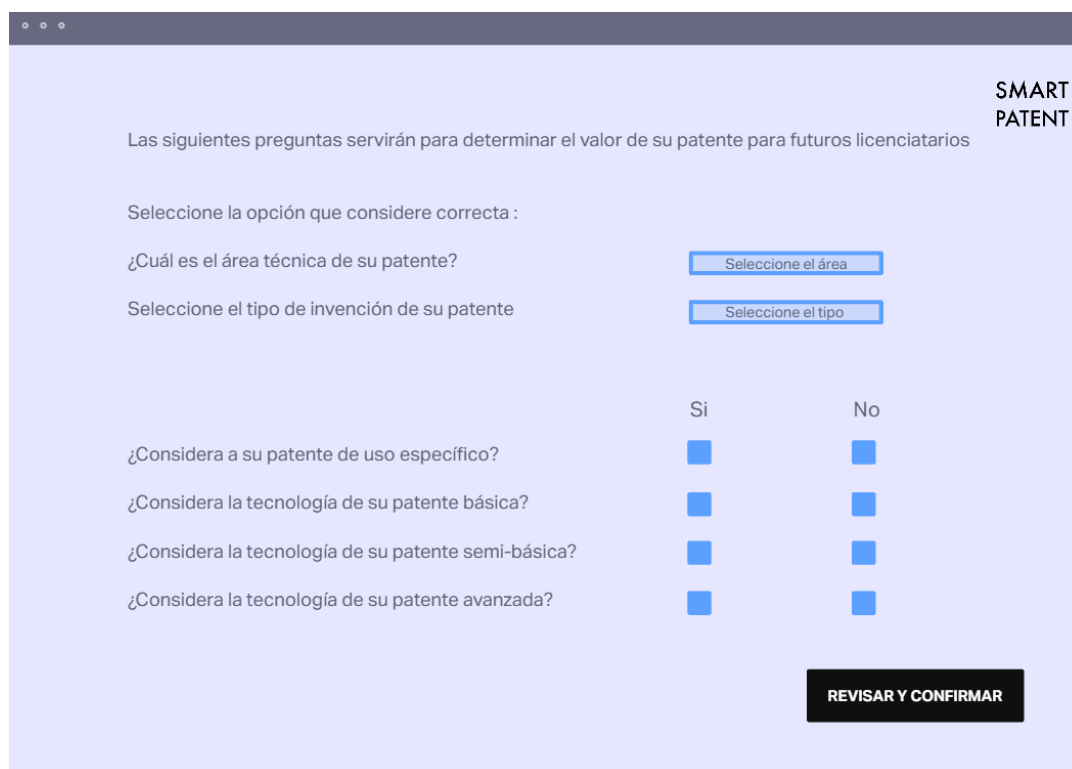
Seleccione la opción que refleje su posición:

¿Cuánto estaría dispuesto a ceder la licencia?

¿En que países quiere ceder su licencia?

¿En que industrias quiere que se comercialice su licencia?

**Figura 23: Interfaz función 3. Elaboración Propia. 2023**



SMART PATENT

Las siguientes preguntas servirán para determinar el valor de su patente para futuros licenciarios

Seleccione la opción que considere correcta :

¿Cuál es el área técnica de su patente?

Seleccione el tipo de invención de su patente

|   | Si                       | No                       |
|---|--------------------------|--------------------------|
| ¿Considera a su patente de uso específico?          | <input type="checkbox"/> | <input type="checkbox"/> |
| ¿Considera la tecnología de su patente básica?      | <input type="checkbox"/> | <input type="checkbox"/> |
| ¿Considera la tecnología de su patente semi-básica? | <input type="checkbox"/> | <input type="checkbox"/> |
| ¿Considera la tecnología de su patente avanzada?    | <input type="checkbox"/> | <input type="checkbox"/> |

**REVISAR Y CONFIRMAR**

**Figura 24: Interfaz función 3. Elaboración Propia. 2023**

La Figura 23, muestra las preguntas a las que tendrá que contestar el licenciante para conocer cómo quiere que sean sus condiciones de licenciatura en caso de estar dispuesto

a licenciar la patente. Después, la Figura 24 nos permite categorizar la patente dentro del sistema para después establecer el precio de esta.

En caso de que la información fuera verificada y aceptada por parte de los validadores, estos ejecutarían el contrato inteligente y este llevaría a cabo la última parte del proceso. Emitiría una transacción quedando registrado todo el proceso incluyendo su etapa final, donde el contrato inteligente acuñaría un NFT que incluiría toda la información referente a la patente y cuyo dueño sería la empresa solicitante. En este caso, la empresa con número identificatorio s5 sería la poseedora de ese NFT.

Entre las múltiples ventajas con las que cuenta esta función respecto al sistema tradicional, destacaría, en primer lugar, la facilidad con la que se agilizan las interacciones futuras con la patente. Gracias al cuestionario que rellena el usuario, el sistema es capaz de establecer de manera prácticamente automática el precio de futuras licencias.

Otra de las ventajas que tendría esta función es que, en este caso, la empresa Attends Healthcare Products Inc podría ir subiendo su proceso de invención por partes, y que esta información quedaría registrada con la fecha de la subida. De esta forma se ahorraría conflictos futuros con los competidores al tener registrado cada paso del proceso. Otra opción es que se quiera mantener en privado la invención, pero que el proceso de registro quede guardado en la red igualmente. En este caso toda la información se haría pública con la transacción final que concedería la licencia de la nueva patente.

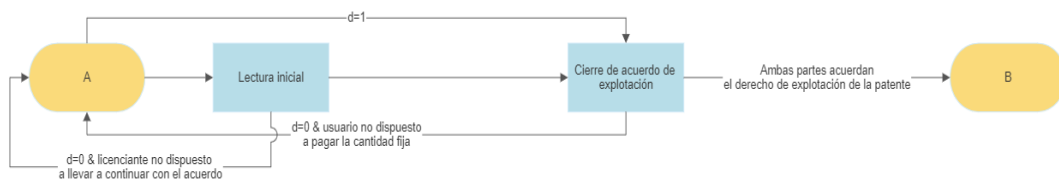
Además, todos los pagos que fueran necesarios debido a que se cumplieran las condiciones prefijadas en el acuerdo se realizarían de manera automática, sin necesidad de costes elevados como sí serían al llevar a cabo procesos burocráticos. El único coste que supondría serían las tarifas de la red de Ethereum, algo ridículamente menor que la cantidad necesaria para pagar estos procesos en el sistema tradicional de patentes.



### 4.3 Caso Práctico de solicitud de licencia de explotación de una patente

El segundo caso práctico del Proyecto explicará el funcionamiento del Modelo a la hora de solicitar la licencia de explotación de una patente. Como se ha explicado en capítulos anteriores, hay determinadas industrias como la tecnológica o la farmacéutica que están en constante desarrollo. Para poder abrirte paso como empresa en el mercado es necesario potenciar la innovación y el desarrollo lo máximo posible. Esta función viene a simplificar ese proceso de innovación, facilitando la obtención de las licencias necesarias para poder patentar un producto de manera legal.

Se introducirá de nuevo el esquema referente a esta función ya que se considera importante para facilitar el seguimiento del caso práctico en cuestión. La Figura 25 (hay que cambiarlo) muestra la estructura general de la función:



**Figura 254: Esquema de la primera función: Solicitud de explotación de una patente. Fuente: Elaboración Propia, 2023**

En la función actual entra un tensor **A** que almacena en sus variables toda la información necesaria para el Modelo, en este caso, el número de licencia de la patente que se solicita, el número identificatorio de la empresa licenciante y una variable booleana para determinar en qué punto del proceso se encuentra. En caso de que el proceso se llevara a cabo (toda la información estuviera en orden y ambas partes llegaran a un acuerdo) se ejecutaría el contrato inteligente. Este estaría permitiendo el acceso al usuario a los metadatos del NFT que contiene toda la información referente a la patente. Por último, el sistema emitiría un tensor **B** que contendría las fechas de validez del acuerdo, las condiciones y la información de la empresa licenciante.

En el caso actual, la empresa Microsoft Technology Licensing Llc. busca obtener la licencia de explotación de la patente con número US9268587B2, esta patente actualmente se encuentra en la posesión de la empresa Empire Technology Development Llc. Entonces, si Microsoft quisiera seguir adelante con la creación de una nueva patente necesitaría tener el derecho de las patentes involucradas en el proceso de creación de esta.

Para este caso práctico, al igual que en los anteriores existen algunos valores que hay que fijar previamente, en este caso esos valores son el número identificatorio de la empresa o el hash de la transacción ejecutada por el contrato inteligente.

Empire Technology Development Llc

- Numero identificatorio ( $s_6$ ): 658957456.
- Dirección en la blockchain de Ethereum:  
0x2175722FbFDe349F4465776FBfB743d342e708c4

Habiendo establecido estos valores, el tensor  $A$ , encargado de almacenar la información de entrada tendría el siguiente aspecto:

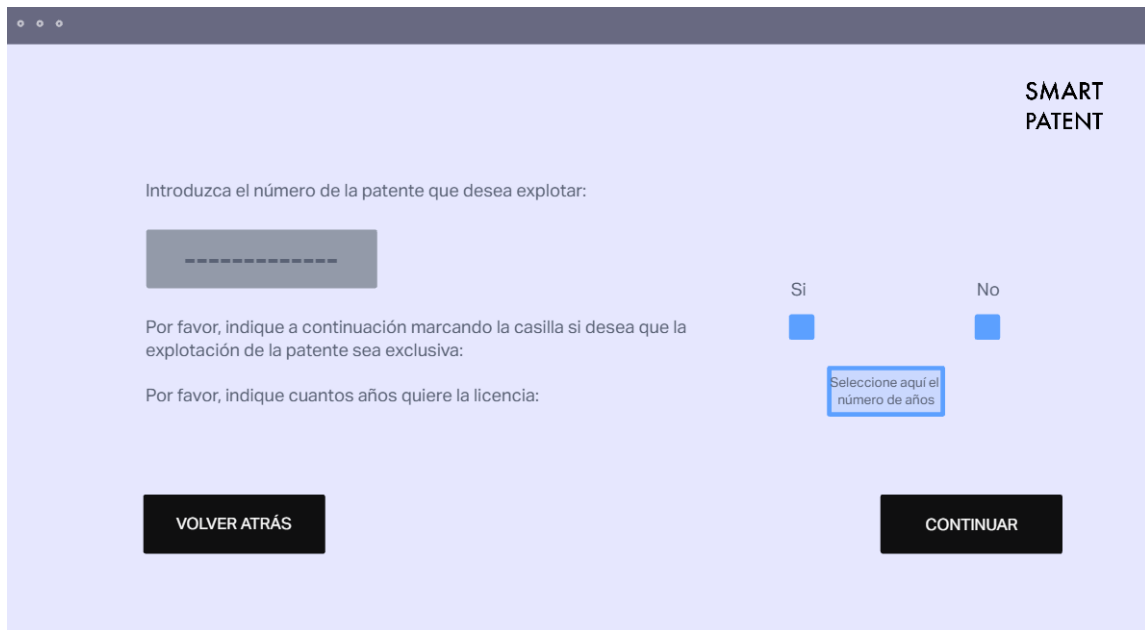
- $a_1$  representa el número de licencia de la patente de interés (US9268587B2).
- $s_6$  representa el numero identificatorio de empresa poseedora de la licencia de la patente (658957456).
- $k$  representa el avance del usuario dentro del proceso del sistema, en este caso asumiremos el valor 0, indicando así que el proceso no ha comenzado

Después, una vez se ha validado la información a la empresa Microsoft, que es la iniciadora de la función, se le presentaría la información de contacto de Empire Technology Development Llc junto con una cantidad monetaria. Si Microsoft no quisiera negociar con  $s_6$ , podría pagar la cantidad directamente por medio de la red de Ethereum y tendría acceso a la patente. En caso de que quisiera negociar y llegar a un acuerdo, contactaría con  $s_6$  y en caso de llegar a un acuerdo, establecerían las nuevas condiciones.

El sistema ejecutaría una transacción donde quedaría registrado el acuerdo por medio del contrato inteligente. Además, emitiría un tensor **B** que recogería toda la información relevante del acuerdo entre sus variables. Las variables del tensor **B** tendrían el siguiente aspecto:

- $a_1$  representa el número de licencia de la patente implicada en el acuerdo (US9268587B2).
- $s_6$  representa el numero identificador de empresa poseedora de la licencia de la patente (658957456).
- $p_t$  representa la fecha de iniciación de la validez del acuerdo de explotación de la licencia (20190115).
- $p_r$  representa la fecha de finalización de la validez del acuerdo de explotación de la licencia (20290223).
- $h_6$  representa el número asociado a la transacción que conlleva el acuerdo en la cadena de bloques de Ethereum:  
0xbf5f087620b3a44a4d9dc1d5748a32efe1b4b6a3f6cf8bb5f9147eb475431406.

A continuación, se mostrarán las distintas pantallas que verá el usuario a la hora de solicitar la licencia de explotación de una patente. Habrá valores que recibirá el sistema, como pueden ser el número identificador de la empresa o la variable booleana  $k$ , que no será necesario que el usuario los introduzca manualmente, irán ligados a su inicio de sesión y solicitud. De esta forma, la Figura 26, muestra cómo se le solicitaría al usuario que introdujera el número de patente que desea explotar, también, si desea que la explotación sea exclusiva o no y por último el número de años que desea solicitar la licencia.



SMART  
PATENT

Introduzca el número de la patente que desea explotar:

-----

Por favor, indique a continuación marcando la casilla si desea que la explotación de la patente sea exclusiva:

Si  No

Por favor, indique cuantos años quiere la licencia:

Seleccione aquí el número de años

VOLVER ATRÁS CONTINUAR

*Figura 26: Interfaz función 1. Elaboración Propia. 2023*

En base a la decisión del futuro licenciatario de explotar la patente de forma exclusiva o no, después, se le mostrarán unas condiciones u otras para la explotación. Por otro lado, en la siguiente Figura 29, se muestra la lista de países que este seleccione dónde quiere explotar la patente.

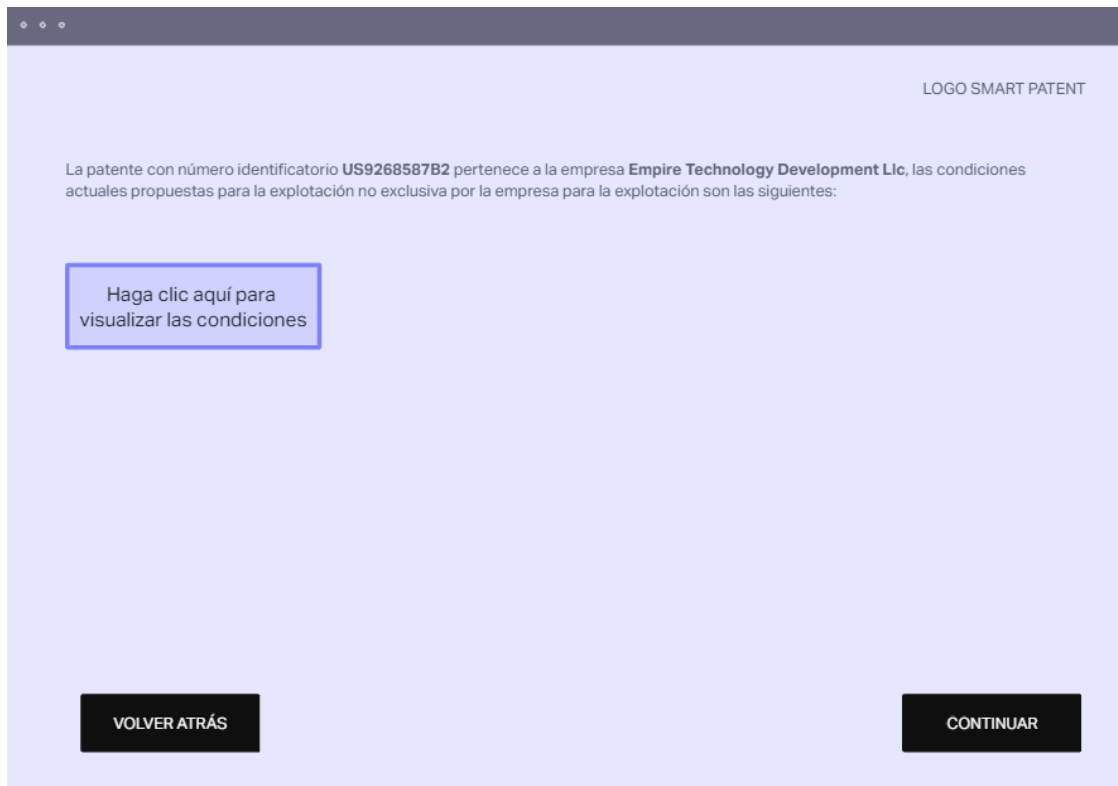
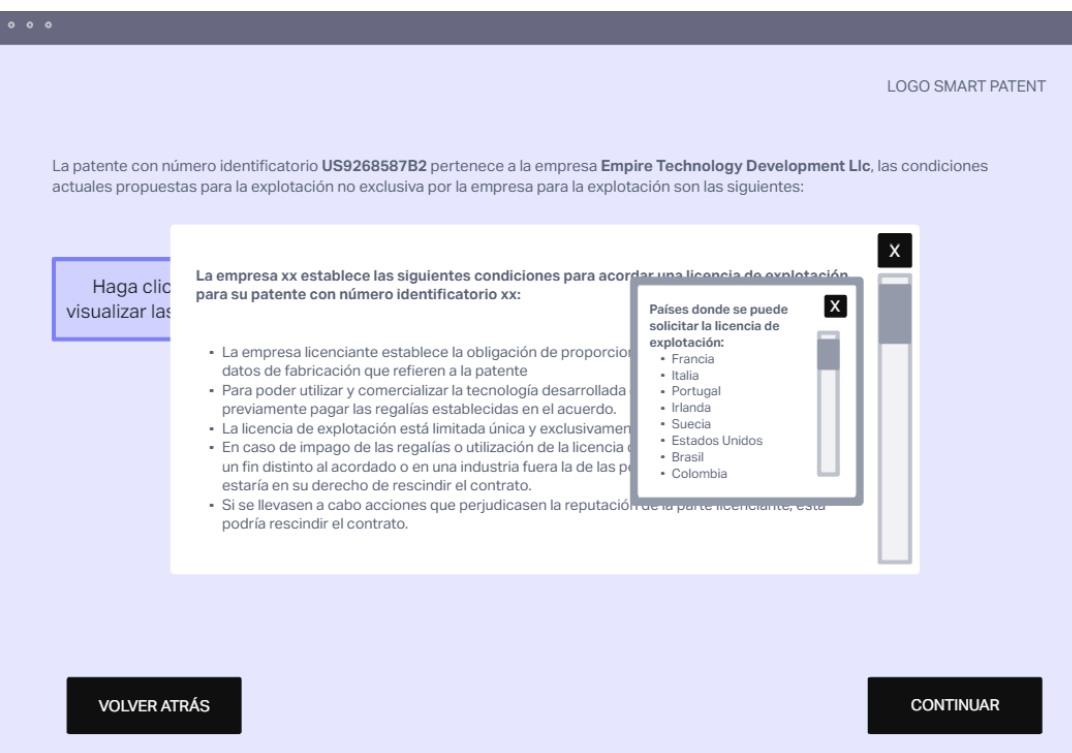


Figura 27: Interfaz función 1. Elaboración Propia. 2023

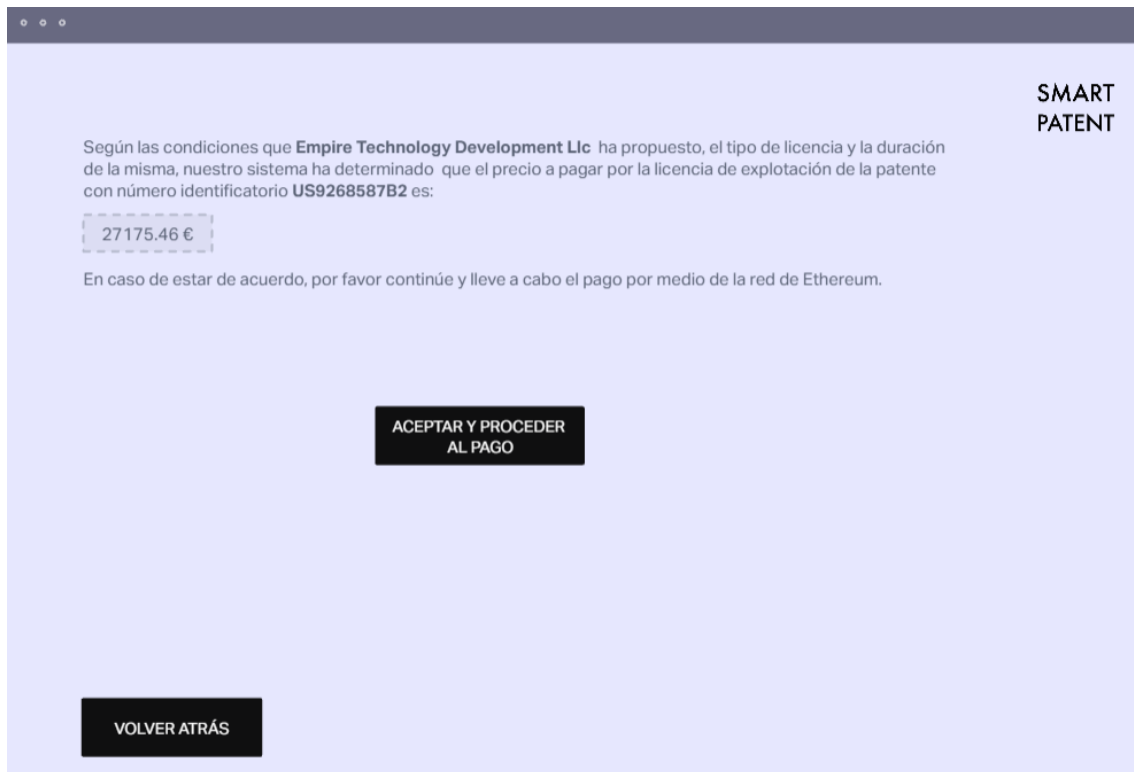


Figura 28: Interfaz función 1. Elaboración Propia. 2023



*Figura 29: Interfaz función 1. Elaboración Propia. 2023*

En las anteriores Figuras 28 y 29, el futuro licenciataria estudiaría las condiciones que se lo proponen para llevar a cabo el acuerdo, así como los países en los que podría solicitar la licencia de la patente. Después, en la Figura 30, se le mostraría al usuario cual sería el precio que debe pagar para obtener la licencia de manera directa. Para este caso, si las condiciones que se fijan son, ajustándonos al caso y estableciendo valores que no salgan de lo común: licencia no exclusiva, duración de diez años encontrándose en el año diez de licencia, tecnología de la patente avanzada, uso no específico, ambas empresas pertenecen a la misma industria y se solicita en cinco de 14 países (35.71%).



*Figura 30: Interfaz función 1. Elaboración Propia. 2023*

En este caso, la cantidad inicial sería la suma de 8916.71 (coste del año en el que se encuentra), a lo que habría que añadirle las tasas de mantenimiento de los diez próximos años (4268.12 euros) y el coste de los agentes de la propiedad (7500 euros). En la figura 18 se observa el precio a pagar por la patente tras haber aplicado los coeficientes (27175.46 euros). El pago se realizaría por medio de la red de Ethereum de manera prácticamente instantánea y automática, habría que cubrir las tasas de la transacción, pero se trata de una cantidad insignificante frente al pago de la licencia.

Por último, en las Figuras 31 y 32 se recogerían la información del acuerdo y se le mostraría al usuario para que, en caso de continuar, estuviera aceptando y haciendo efectivo el acuerdo.



*Figura 31: Interfaz función 1. Elaboración Propia. 2023*



*Figura 32: Interfaz función 1. Elaboración Propia. 2023*



Según las condiciones actuales, el razonamiento expuesto en el anterior capítulo indicaría que el precio a pagar por esa licencia de patente sería de 27175.46 €. De esta forma, se habría conseguido prácticamente automatizar este proceso gracias a las respuestas del licenciante cuando se creó la patente, la añadió al sistema e impuso sus condiciones frente a futuras licencias.

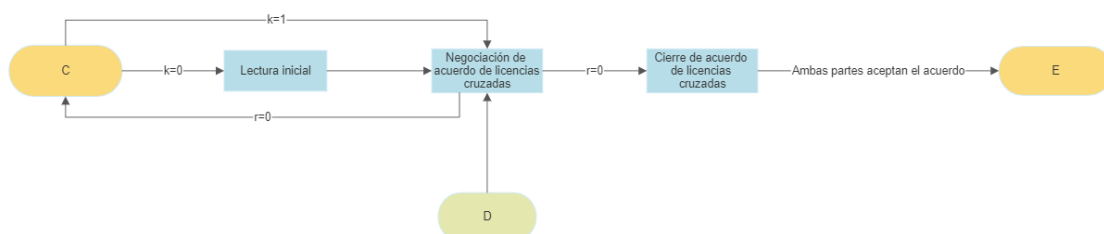
Gracias a esta función se facilitaría el pago de royalties al licenciario, al estar las condiciones del acuerdo incluidas en un contrato inteligente estos pagos se realizarían de manera automática. Por otro lado, la opción pagar la cantidad ya fijada por el licenciario ahorraría tiempo en caso de este ser necesario.

También, en esta función se pueden apreciar la simplificación que aporta al proceso el haber automatizado el cálculo del valor de la patente según el Modelo planteado. Esta función no deja de ser la más sencilla de todas las expuestas, pero, aun así, refleja las ventajas de realizar todos estos procesos incluyendo las características de la tecnología blockchain.

## 4.4 Casos Prácticos de acuerdos de licencias cruzadas

En este subcapítulo pondremos en práctica dos situaciones reales de acuerdos de licencias cruzadas entre empresas. Para el primero de los casos las empresas en cuestión serán Samsung y Cisco, por otro lado, el segundo será entre Ericsson y Apple. En ambos casos, como es común en los acuerdos de licencias cruzadas todas las empresas apuestan por la innovación, ya sea patentando un nuevo producto o proceso, para ello necesitan tener el derecho de explotación de todas licencias de las patentes implicadas en el proceso de elaboración de lo que se quiere patentar, por lo que es de su interés firmar este acuerdo.

Se considera importante para la correcta comprensión del caso práctico comentar brevemente la estructura que muestra la Figura 33 a continuación, donde se puede ver el funcionamiento general de la función encargada de gestionar acuerdos de licencias cruzadas.



**Figura 33: Esquema de la segunda función: Acuerdo de licencias cruzadas. Fuente: Elaboración Propia, 2023**

En la anterior figura se observa un tensor de entrada **C**, encargado de recopilar la información de entrada del acuerdo, un tensor **D**, encargado de introducir en el sistema la información proporcionada por la empresa no iniciadora del acuerdo. Por último, una vez ambas partes firman el acuerdo, un tensor de salida **E** responsable de recopilar toda la información referente a las condiciones pactadas por ambas empresas. Además, el contrato inteligente que gestiona esta función emitiría un NFT para cada empresa que le concedería a cada una el acceso a las patentes incluidas en el acuerdo.

En ambos casos se seguirá una estructura similar a la hora de poner en práctica el modelo. En primer lugar, se explicará la situación de cada empresa, especificando las patentes que son de su interés. Después se adaptarán los datos de la situación real a las componentes de los tensores **C** y **D**. Por último, se explicará el proceso seguido a lo largo de la función de solicitar acuerdos de licencias cruzadas y se extraerán unas conclusiones de este.

Después, en el primer caso, se presentarán las distintas pantallas que observará el usuario al interactuar con nuestro sistema en la función “Acuerdos de Licencias Cruzadas”. También, se explicará el razonamiento lógico detrás del precio ofrecido al usuario para asegurar la correcta comprensión del modelo. Para concluir se realizarán unos breves comentarios sobre de la función expuesta.

#### **4.2.1 Primer Caso Práctico de acuerdos de licencias cruzadas**

En este primer caso, la empresa Cisco Technology busca obtener la licencia para patentar “Aprendizaje de reglas sólidas y precisas para la clasificación de dispositivos a partir de grupos de dispositivos”, para ello necesitaría obtener previamente la licencia de explotación de la patente con número US20180114123A1, el licenciataria de esta patente es Samsung Electronics Co. Entonces para Cisco Technology es de su interés llegar a un acuerdo con Samsung, ya sea acordando una cantidad económica que pagar o mediante un “intercambio de licencias de patentes” por medio de un acuerdo de licencias cruzadas.

Por otro lado, observamos que Samsung busca patentar “Sistema integrado de telefonía por Internet y método de señalización del mismo”, pero al igual que Cisco Technology necesita, en este caso dos patentes de la otra empresa implicada en el acuerdo, US20050195811A1 y US20080159306A1.

En esta situación se observa que una clara solución sería un acuerdo de licencias cruzadas entre ambas empresas. Aplicándolo a nuestro modelo, habría que establecer a una de las dos empresas como iniciadora del proceso necesitado para llevar a cabo la negociación del acuerdo de licencias cruzadas.

Para el primer caso de aplicación práctica, se establecerá que la empresa iniciadora es Samsung, por lo que será la encargada de rellenar la información de las variables del tensor de entrada **A**.

En primer lugar, para poder utilizar el modelo, habrá que adaptar la situación real a los datos necesarios del modelo. Para ello, deberemos asociar a cada empresa un número identificador, así como una dirección en la blockchain, a la que después, en caso de llevar a cabo el acuerdo, se le concederá acceso a la información de las patentes.

Entonces, para este primer caso, la información asociada a cada una de las empresas será la siguiente.

Samsung Electronics Co.

- Numero identificador( $s_1$ ): 758548695.
- Dirección en la blockchain de Ethereum:  
`0xC7Ea7cF34AD50275c055dCfF0b6E80D94E45B7ca`

Cisco Technology

- Numero identificador ( $s_2$ ): 115665335.
- Dirección en la blockchain de Ethereum:  
`0xc99e2F42622863149D078eb5710d9410541e9D2E`

Habiendo establecido los números y direcciones ahora tendremos que completar toda la información del tensor de entrada **C**. Primero se expondrá sus valores y después se explicará el porqué de esos valores.

Las variables del tensor **C** son las siguientes:

- $q_1$ , cuyas componentes del vector serán los numero de las licencias de las patentes de interés para Samsung (US20050195811A1 y US20080159306A1).
- $s_2$ , el número identificador de Cisco (115665335).
- $s_1$ , el número identificador de Samsung (758548695).
- $c_1$ , las condiciones propuestas por Samsung para llevar a cabo el acuerdo.
- $k$ , en este caso asumiremos el valor 0, indicando que el proceso no ha sido empezado previamente.

En este caso,  $s_2$  será Cisco, que cuenta con las licencias de las patentes de interés para Samsung ( $s_1$ ). La variable  $c_1$ , que será un archivo de texto, incluirá en su interior las condiciones propuestas por Samsung. En este caso, las condiciones podrían ser las siguientes:

- Solicitar el derecho de explotación de la patente de interés teniendo en cuenta que la licencia expira a los 20 años de la concesión de esta.
- Acordar la territorialidad de esta, es decir, en que regiones tiene validez este acuerdo de licencias cruzadas
- Acordar los royalties teniendo en cuenta factores como el número de licencias intercambiadas, la territorialidad o el tiempo hasta la caducidad de las patentes.

Después, el sistema validaría si toda la información es correcta, en caso de serlo, se le proporcionaría a la empresa  $s_1$  la información de contacto de  $s_2$ . Una vez se han puesto en contacto, la empresa con número identificadorio  $s_2$  entraría al sistema con sus nuevas condiciones propuestas para cerrar el acuerdo y establecer las patentes que son de su interés.

Entonces, el tensor encargado de cargar en el sistema la información referente a la empresa  $s_2$ ,  $\mathbf{D}$ , estará formado por las siguientes variables, cuyos valores serán expuestos a continuación:

- $q_2$ , cuyas componentes serán los números de licencias de las patentes de interés para Cisco (US20090028138A1).
- $s_2$ , el número identificadorio de Cisco (115665335).
- $c_2$ , las nuevas condiciones propuestas por Cisco, necesarias para cerrar el acuerdo.
- $r$ , supondremos que el valor de esta es de 1, indicando así que la empresa  $s_2$  está dispuesta a seguir con el acuerdo.

Las nuevas condiciones de la empresa  $s_2$ , podrían incluir también una compensación económica por parte de  $s_1$ , al no ser el número de patentes intercambiadas el mismo. También se tendrían en cuenta los aspectos mencionados en las condiciones propuestas

por la empresa  $s_1$ . Por otro lado, también se incluirían unas cláusulas por las que se rompería el acuerdo, por ejemplo:

- Incumplimiento de pagos de los royalties por una de las partes.
- Cambio de condiciones en caso de que una de las dos partes fuera adquirida por otra empresa.

Al estar todo el modelo basado en la interacción con un contrato inteligente, toda la información del acuerdo estaría incluida dentro de la cadena de bloques de la red de Ethereum, y al ejecutarse el acuerdo e interactuar con el contrato inteligente toda la información quedaría registrada en la blockchain. Esta transacción quedará registrada por nuestro modelo, formando parte del tensor de salida  $E$ .

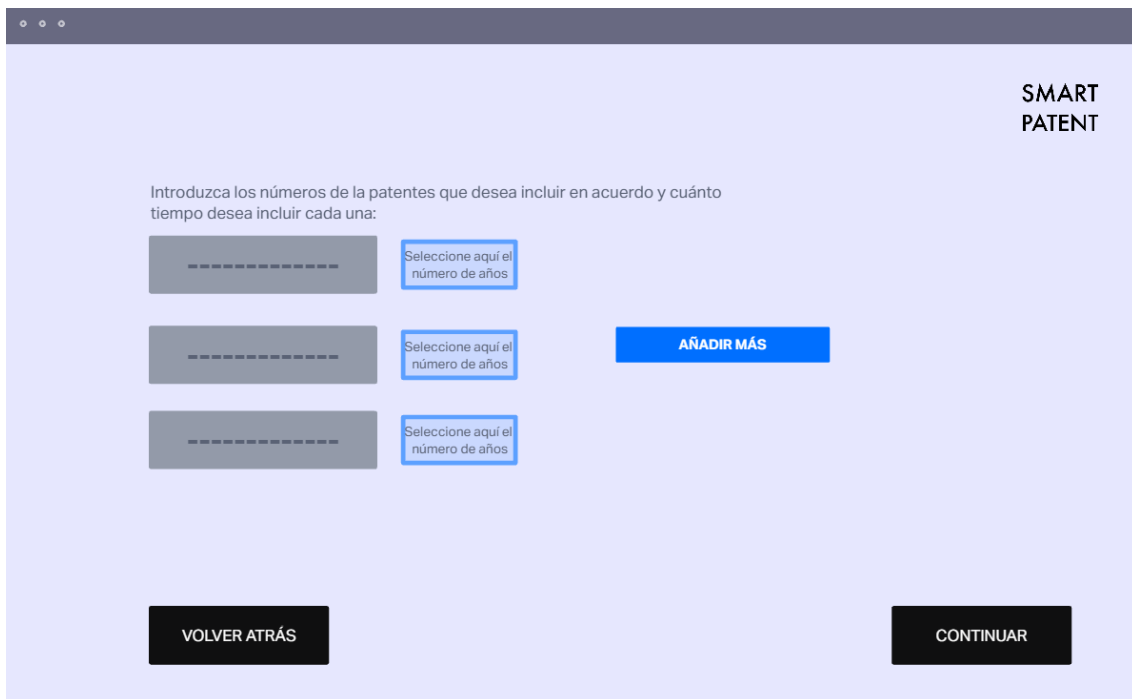
Este tensor contará con toda la información del acuerdo agrupada en distintas variables, todas ellas conforman el tensor  $E$ . Los valores de las variables de este tensor podrían ser los siguientes:

- $p_{12}$ , vector con las licencias de patentes implicadas en el acuerdo (US20050195811A1, US20080159306A1 y US20090028138A1).
- $s_1$ , número identificador de Samsung (758548695).
- $s_2$ , número identificador de Cisco (115665335).
- $p_1$ , fecha de iniciación del acuerdo entre las empresas  $s_1$  y  $s_2$  (20170101).
- $p_2$ , fecha de finalización del acuerdo entre las empresas  $s_1$  y  $s_2$  (20350101)
- $c_{12}$ , condiciones acordadas y aceptadas por ambas empresas.
- $h_{12}$ , hash asociado a la transacción del acuerdo de licencias cruzadas en la cadena de bloques:  
0x63a48278efae487286f00d94784daa92726f79974238585a51a8528d28d8b798.

Por último el contrato inteligente, emitirá un NFT para cada empresa que servirá como certificado digital a cada empresa, de esta forma, si la empresa con número identificador  $s_1$  quisiera acceder a la información de la patente US20180114123A1 el sistema leería que la dirección 0xc99e2F42622863149D078eb5710d9410541e9D2E

posee un NFT que acredita su acceso y se le concedería permiso para la lectura de toda la información referente a esa patente.

A continuación, se mostrarán capturas y explicaciones de cómo sería el proceso para ambas partes dentro de un acuerdo de licencias cruzadas. En primer lugar, la Figura 34, muestra la pantalla que vería la empresa interesada en empezar el acuerdo de licencias cruzadas, donde se le pediría que incluyera los números de las patentes de interés, así como los años que desea incluirlas en el acuerdo.



The screenshot shows a web interface titled "SMART PATENT". The main heading is "Introduzca los números de la patentes que desea incluir en acuerdo y cuánto tiempo desea incluir cada una:". Below this, there are three rows, each consisting of a grey input field for the patent number and a blue dropdown menu labeled "Seleccione aquí el número de años". To the right of these rows is a blue button labeled "AÑADIR MÁS". At the bottom of the form are two black buttons: "VOLVER ATRÁS" on the left and "CONTINUAR" on the right.

*Figura 34: Interfaz función 2. Elaboración Propia. 2023*

Después, la empresa iniciadora del acuerdo, en este caso Samsung, subiría sus condiciones del acuerdo de licencias cruzadas al sistema, como se puede apreciar en la Figura 35. A continuación, también se le presentaría una pantalla de confirmación para verificar que la información es correcta, como se puede observar en la Figura 36.



Figura 5: Interfaz función 2. Elaboración Propia. 2023

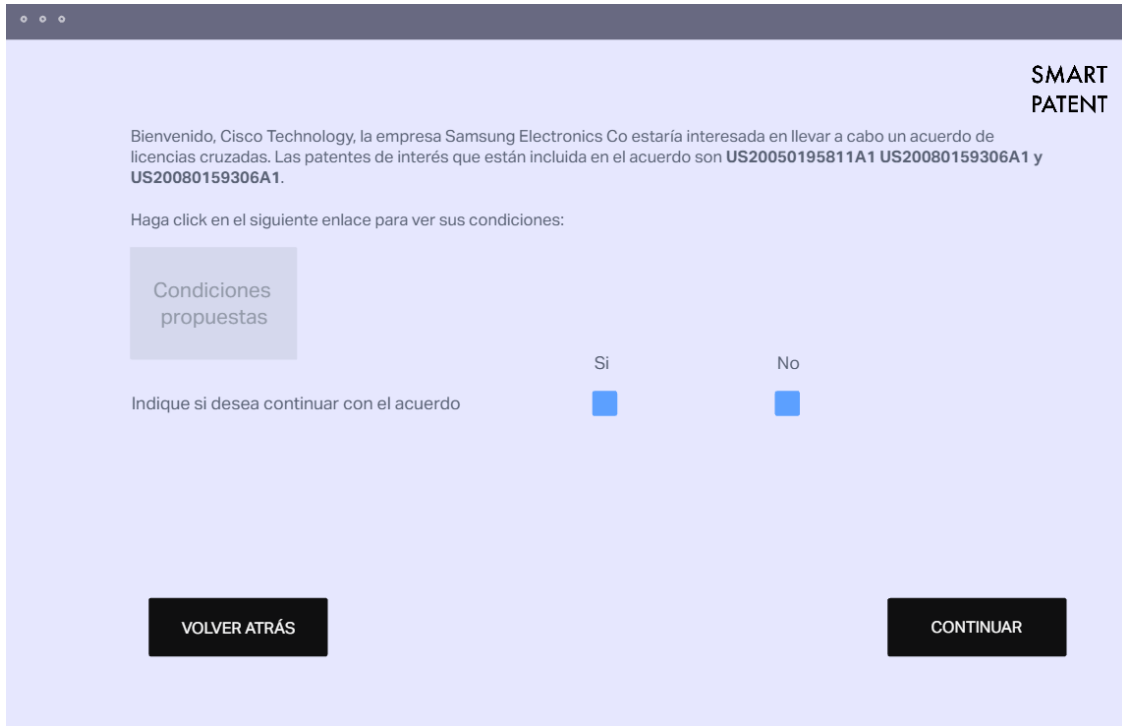


Figura 36: Interfaz función 2. Elaboración Propia. 2023

En esta parte, entraría en el sistema aquella empresa no iniciadora implicada en el acuerdo, donde se le mostraría la siguiente pantalla de bienvenida, así como las



condiciones que han sido propuestas previamente por la otra empresa, en este caso Samsung.



SMART  
PATENT

Bienvenido, Cisco Technology, la empresa Samsung Electronics Co estaría interesada en llevar a cabo un acuerdo de licencias cruzadas. Las patentes de interés que están incluida en el acuerdo son **US20050195811A1 US20080159306A1 y US20080159306A1**.

Haga click en el siguiente enlace para ver sus condiciones:

Condiciones propuestas

Indique si desea continuar con el acuerdo

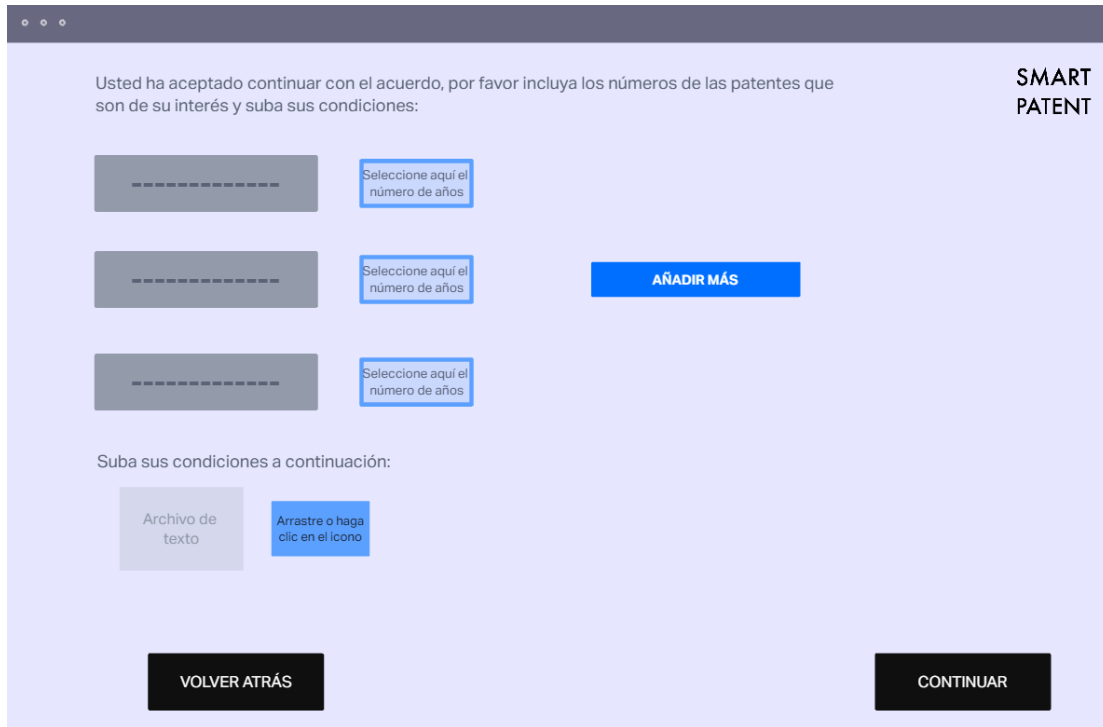
Si

No

VOLVER ATRÁS CONTINUAR

*Figura 37: Interfaz función 2. Elaboración Propia. 2023*

Para el caso práctico en cuestión supondremos que Cisco aceptaría el acuerdo y pasaría a mostrarse la siguiente Figura 38 donde tendría que incluir las patentes de su interés, así como el tiempo que desea explotarlas.



The screenshot shows a web interface for 'SMART PATENT'. At the top right, the text 'SMART PATENT' is displayed. The main content area contains a message: 'Usted ha aceptado continuar con el acuerdo, por favor incluya los números de las patentes que son de su interés y suba sus condiciones:'. Below this, there are three rows, each consisting of a grey dashed input field and a blue button labeled 'Seleccione aquí el número de años'. To the right of these rows is a blue button labeled 'AÑADIR MÁS'. Below the input fields, there is a section titled 'Suba sus condiciones a continuación:' which includes a grey box labeled 'Archivo de texto' and a blue button labeled 'Arrastre o haga clic en el icono'. At the bottom of the interface, there are two black buttons: 'VOLVER ATRÁS' on the left and 'CONTINUAR' on the right.

*Figura 38: Interfaz función 2. Elaboración Propia. 2023*

Después de esto, el sistema procedería a calcular cuál sería el precio objetivo de cada licencia de patente en base a las preguntas que respondió cada licenciante cuando introdujeron las patentes al sistema. Entonces, la cantidad monetaria que deberá pagar cada empresa en caso de ser necesario sería la diferencia entre la suma del precio calculado por nuestro sistema de todas las patentes para Samsung y para Cisco. Al ser un acuerdo de licencias cruzadas, el coeficiente que refleja si dos empresas pertenecen a la misma industria y por tanto deberá rebajarse el precio final con el fin de promover la innovación y el desarrollo, en la mayoría de los casos tendrá el valor de 0.8.

En este caso de licencias cruzadas, para simplificar el cálculo final del precio de todas las patentes se establecerán las siguientes condiciones que influyen tanto en la cantidad inicial  $c$  como en los distintos factores que multiplican a esta:

- Todas las licencias, al tratarse de empresas tecnológicas con gran presencia internacional, se supondrán de tecnología avanzada, por lo que  $t$  tendrá el valor 1.1.

- Se asumirá que las licencias que se incluyen en este acuerdo no podrán concederse a otros licenciantes, es decir, se tratarán como exclusivas. Es lógico pensar que esto lleve a cabo de esta forma por el beneficio de ambas empresas como competidores en la industria tecnológica. Entonces **e** tendrá el valor 1.1.
- La duración de interés para ambas empresas será de tres años, pero ambas no se encontrarán en el mismo punto con respecto a su licencia, es decir, se desarrollará en que punto de la licencia se encuentra cada una para valorar el coste de la licencia. Por ello, **d** tendrá el valor de 1.15.
- En ambos casos, el acuerdo será válido en todos los países de las patentes donde cada empresa sea dueña de la patente. Tiene sentido que si una patente se pide en países de mayor inversión tecnológica se solicite también en el resto de los países con el fin de ser competitivos en el mayor territorio posible. Entonces, **n** tendrá el valor 2.
- Las patentes de las dos empresas se considerarán de uso específico, al tratarse de procesos tecnológicos con un gran desarrollo en su trabajo previo. Por ello, **u** tendrá el valor de 0.9.

Entonces, una vez se han establecido estos parámetros, se procede a enseñar la siguiente captura, donde el usuario verá el precio de cada patente calculado por nuestro sistema, así como la suma total. Además, también aparecerán las cláusulas finales del acuerdo, así como la cantidad que tenga que pagar cada empresa.



**SMART  
PATENT**

En el presente acuerdo de licencias cruzadas la empresa **Samsung** deberá pagar la cantidad de **39386.95 €** para llevarlo a cabo.

El valor asociado a cada patente y como el computo global aparecen indicados en la siguiente tabla:

|         | Patentes de interés | Precio Final | Precio Total | Cantidad a Pagar |
|---------|---------------------|--------------|--------------|------------------|
| Samsung | US20050195811A1     | 36104.79     | 72372.61     | 39386.95         |
|         | US20080159306A1     | 36267.82     |              |                  |
| Cisco   | US20180114123A1     | 32985.66     | 32985.66     | 0                |

VOLVER ATRÁS
CONTINUAR Y PROCEDER CON EL PAGO

*Figura 396: Interfaz función 2. Elaboración Propia. 2023*

El cálculo detrás del precio final de las licencias es el siguiente:

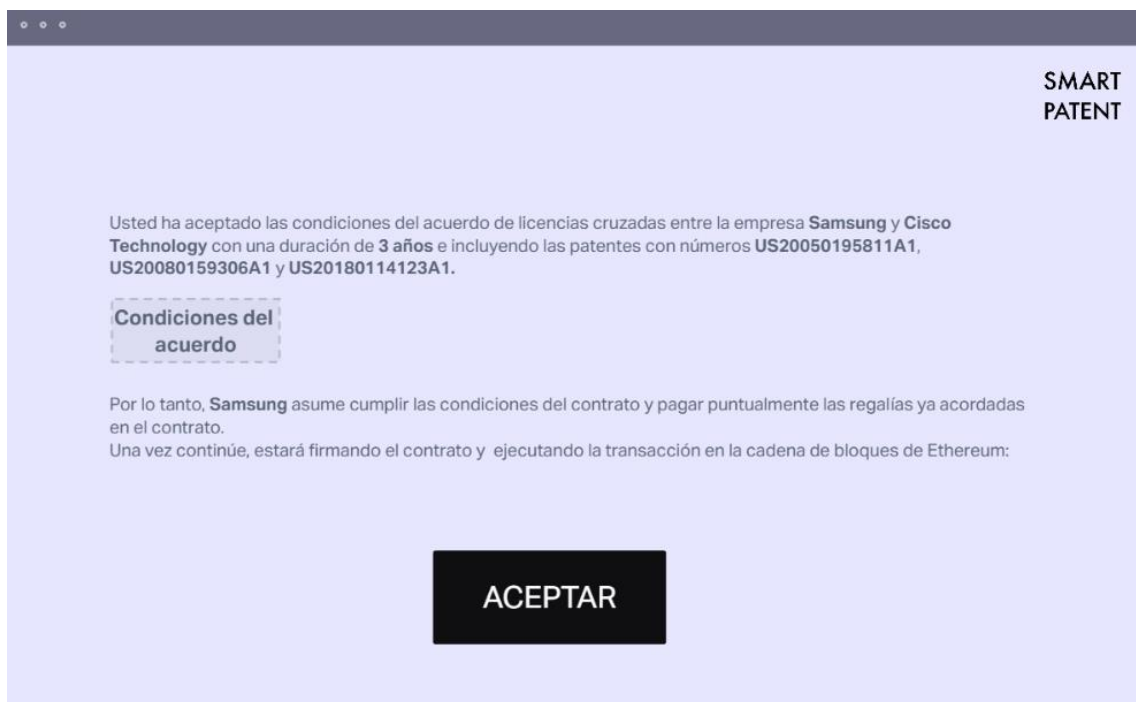
| Empresa | Número de las patentes de interés | Precio actual (€) | Tasas próximos años (€) | Precio abogado(€) | Coefficientes(€) | Cantidad inicial(€) | Precio final (€) | Valor total licencias (€) |
|---------|-----------------------------------|-------------------|-------------------------|-------------------|------------------|---------------------|------------------|---------------------------|
| Samsung | US20050195811A1                   | 9193.39           | 1325.13                 | 7500              | 2.004            | 18018.52            | 36104.79         | 72372.61                  |
|         | US20080159306A1                   | 9115.18           | 1484.7                  | 7500              | 2.004            | 18099.88            | 36267.82         |                           |
| Cisco   | US20180114123A1                   | 8743.04           | 218.84                  | 7500              | 2.004            | 16461.88            | 32985.66         | 32985.66                  |

*Figura 40: Cálculo detrás del acuerdo de licencias cruzadas en la función 2. Elaboración Propia. 2023*

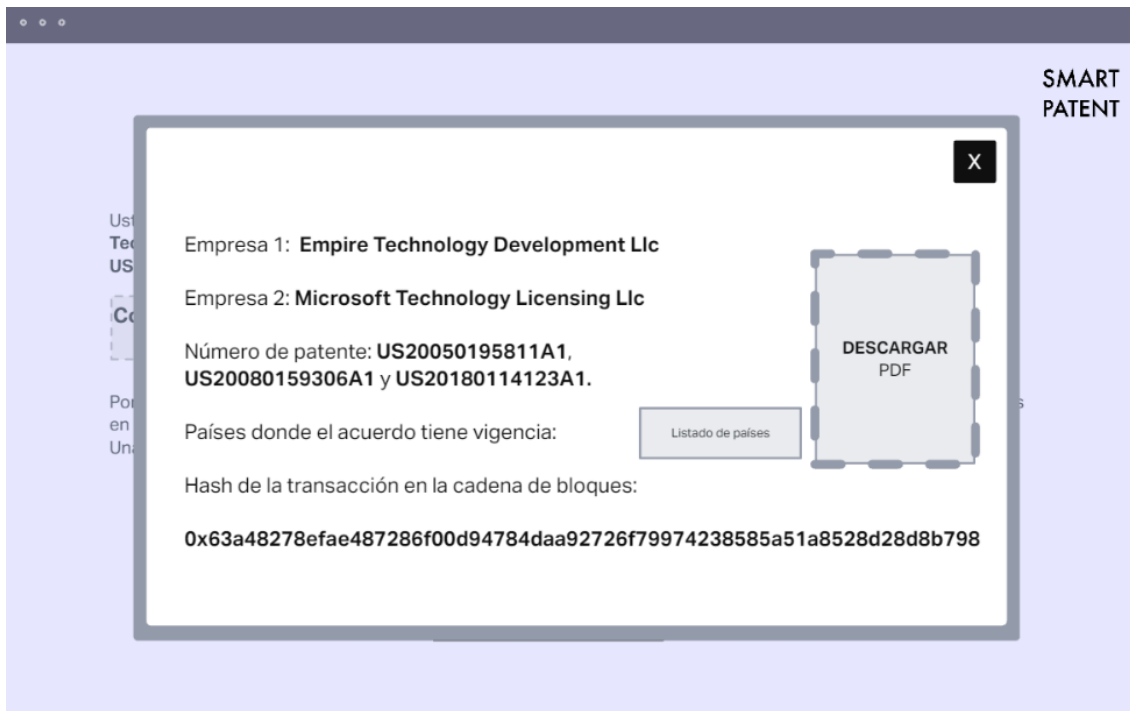
Como se puede apreciar en la Figura 40, el precio actual de cada licencia es distinto, y depende del año de la licencia en el que se encuentre, en el caso de las licencias en posesión de Cisco Technology, una está en el decimoséptimo año de licencia y la otra en el decimocuarto, por ello al ser las tasas mayores, la cantidad inicial es mayor. Por otro lado, en el caso de Samsung, sus licencias se encuentran en el quinto año, por lo que las tasas serán menores.

Al final del todo, la cantidad que le aparece a pagar a Samsung es la diferencia entre el valor de sus patentes de interés y las de Cisco Technology, en este caso la diferencia es de 20678.15 € y el pago se realizaría por medio de la red de Ethereum.

Por último, en la Figura 41 y 42, se puede apreciar la captura de confirmación del acuerdo que le aparecería al usuario, en este caso Samsung. En esa pantalla, el usuario podría visualizar las condiciones del acuerdo, así como las fechas de inicio y fin y las patentes incluidas en este. Por último, le aparecería una ventana de confirmación donde iría incluido el nombre ambas empresas, los números de las patentes implicadas, las condiciones y duración del acuerdo en un PDF que podría descargarse, la territorialidad del acuerdo y el hash asociado a la transacción emitida por el contrato inteligente.



*Figura 417: Interfaz función 2. Elaboración Propia. 2023*



*Figura 42: Interfaz función 2. Elaboración Propia. 2023*

#### 4.2.2 Segundo Caso Práctico de acuerdos de licencias cruzadas

En este segundo caso práctico las empresas implicadas son Ericsson y Apple, ambas empresas cuentan con una importante presencia dentro de la industria telefónica por lo que un acuerdo de licencias cruzadas entre ellas es lo lógico si se quiere apostar por la innovación y el desarrollo. En este caso, la empresa Apple Inc. busca obtener la licencia de “Mapeo de un canal físico de control de enlace descendente Mejorado”, para ello, previamente necesitaría obtener las licencias de explotación de las patentes que intervengan en el proceso que se quiere patentar con el fin de no incumplir ningún proceso legal. Además, gran parte de estas licencias de patentes se encuentran en la posesión de Telefonaktiebolaget L M Ericsson por lo que Apple Inc. estaría interesada en este acuerdo. Por otro lado, Telefonaktiebolaget L M Ericsson busca patentar “Gestión de llamadas de voz en cola” pero, de nuevo, entre las licencias de explotación que necesita se encuentra alguna cuyo propietario es Apple Inc. Entonces, por todo lo

previamente explicado un acuerdo de licencias cruzadas entre ambas empresas sería un proceso legal apropiado para esta situación.

Al igual que en el caso anterior, en primer lugar, se establecerán algunos valores de las variables de los tensores **C**, **D** y **E** para que el modelo quede completo en cuanto a datos. Estos valores iniciales se fijan con el objetivo de acercar al máximo el caso práctico del modelo con la realidad. En este caso, consideraremos que Apple es la empresa iniciadora del acuerdo ya que es la que necesita un mayor número de patentes.

Los valores fijados inicialmente serán los siguientes:

Apple Inc.

- Numero identificador( $s_3$ ): 568425958.
- Dirección en la blockchain de Ethereum:  
0x51a47ebaBA37121577C96Bbf1bd80a113af1E86f

Telefonaktiebolaget L M Ericsson

- Numero identificador ( $s_4$ ): 785649215.
- Dirección en la blockchain de Ethereum:  
0xe484243fc575F45a06d7183c1eB34fA71FB352A3

Una vez se han establecido estos valores ya podemos completar como quedaría el tensor **C**, encargado de almacenar en sus variables la información proporcionada por la empresa iniciadora del acuerdo. El tensor **C** en cuestión contaría con los siguientes valores dentro de sus variables:

- $q_2$ , cuyas componentes del vector serán los numero de las licencias de las patentes de interés para Apple (US8285346B2, US9247501B2 y US8761093B2).
- $s_3$ , el número identificador de Apple (568425958).
- $s_4$ , el número identificador de Ericsson (785649215).
- $c_3$ , las condiciones propuestas por Apple para llevar a cabo el acuerdo de licencias cruzadas.

- $k$ , en este caso asumiremos el valor 0, indicando que el proceso no ha sido empezado previamente.

En este caso, entre las condiciones propuestas por Apple podrían estar incluidas las regalías, la territorialidad de las patentes o limitaciones en la competencia. De esta forma, Apple conseguiría fortalecer su posición en ciertos mercados. Además, también buscaría que el acuerdo se prolongara el máximo tiempo posible mientras las licencias se encuentren en la posesión de Ericsson.

El sistema después verificaría la información proporcionada por Apple y validaría el comienzo de la solicitud proporcionándole la información de contacto de Ericsson. Una vez ambas empresas han contactado, Ericsson accedería al sistema por medio del tensor  $D$  donde plantearía sus nuevas condiciones en caso de tenerlas e incluiría las patentes que son de su interés. Entonces, el tensor  $D$  tendría los siguientes valores entre sus variables:

- $q_2$ , cuyas componentes serán los números de licencias de las patentes de interés para Ericsson (US8363818B2).
- $s_4$ , el número identificador de Ericsson (115665335).
- $c_2$ , las nuevas condiciones propuestas por Ericsson, necesarias para cerrar el acuerdo.
- $r$ , supondremos que el valor de esta es de 1, indicando así que la empresa  $s_4$  está dispuesta a seguir con el acuerdo.

Dentro de las condiciones propuestas por Ericsson tendría sentido incluir una cantidad inicial a pagar por parte de Apple, ya que el número de patentes intercambiado no es el mismo. Además, también podría especificar la territorialidad y la duración del acuerdo en base a sus intereses como empresa.

Después de esto, cada empresa valoraría su situación actual y los cambios que conllevarían firmar el acuerdo bajo las condiciones aceptadas por ambas partes. Con la intención de analizar al máximo el modelo, se considerará que ambas partes aceptan el presente acuerdo.

Entonces, el tensor de salida  $E$  contendría la siguiente información:



- $p_{34}$ , vector con las licencias de patentes implicadas en el acuerdo (US8285346B2, US9247501B2, US8761093B2 y US8363818B2).
- $s_3$ , número identificador de Apple (568425958).
- $s_4$ , número identificador de Ericsson (785649215).
- $p_3$ , fecha de iniciación del acuerdo entre las empresas  $s_3$  y  $s_4$  (20180101).
- $p_4$ , fecha de finalización del acuerdo entre las empresas  $s_3$  y  $s_4$  (20350101)
- $c_{34}$ , condiciones acordadas y aceptadas por ambas empresas.
- $h_{34}$ , hash asociado a la transacción del acuerdo de licencias cruzadas en la cadena de bloques:

0xfbed6c03b8f4be4d9bf2c11cc73a375277680794ffe33ff9b8bfb8da19f14893

Por último, una vez ambas partes han firmado el acuerdo, el sistema emitiría un NFT que dotaría de acceso a la información de las patentes a cada empresa implicada. Esta emisión del token no fungible la realizaría el contrato inteligente encargado de gestionar el acuerdo de licencias cruzadas entre Apple y Ericsson. Todo este proceso se estaría llevando a cabo de manera descentralizada y prácticamente automatizada por código, se estaría ahorrando tiempo y dinero a la hora de desarrollar este acuerdo, aunque esto último se discutirá más adelante en el siguiente capítulo.

Se podría considerar que para esta función el modelo alcanza su máxima potencia al estar simplificando muchas más gestiones e interacciones que en las otras funciones. Ciertamente es también que el código que automatiza todo este proceso es el más complejo de todos y a la hora de programarlo específicamente para cada acuerdo sea este el que más inversión de tiempo y dinero requiera. De todas formas, el Modelo aporta muchas soluciones frente al sistema actual, una de ellas podría ser la comodidad a la hora de firmar, sin necesidad de notarios. Otra podría ser la capacidad para modificar alguna cláusula, siempre y cuando ambas partes estén de acuerdo, sin la obligación de la presencialidad de las empresas y todo con la firma criptográfica de la dirección en la blockchain.

## 4.5 Conclusiones

Al observar el funcionamiento de los casos prácticos, a simple vista se puede observar que habría un ahorro de tiempo gracias a que el proceso estaría automatizado. Todo el proceso se llevaría a cabo de manera remota, sin necesidad de la presencialidad para ningún pago o firma, ya que se estaría realizando todo por medio de la tecnología blockchain. Además, uno no dependería de un tercero para que firmase las transacciones para después realizar los pagos, ya que cada uno es responsable de su dirección en la blockchain.

En los casos de acuerdos de licencias cruzadas, al poder establecer la territorialidad del acuerdo y que este quede reflejado en la cadena de bloques, esto hace que las condiciones de este puedan ser consultadas desde cualquier parte del mundo. Gracias a esto último podemos evitar posibles conflictos que podrían ocurrir con el sistema tradicional al no estar toda la información digitalizada o al alcance de todo el mundo en cualquier momento. Además, el hecho de que dos empresas tecnológicas como son Apple y Ericsson un acuerdo de licencias cruzadas, es sinónimo de apostar y promover por la innovación y bienestar de todos, ya que, simplificándolas este proceso, son capaces de invertir más tiempo y dinero en I+D.

También, por otro lado, en el caso de la patente “Artículo absorbente con núcleo absorbente plegado de capa múltiple”, al realizarse todo de manera remota aumenta la comodidad del proceso. Evitando así esperas y pudiendo consultar el avance de la solicitud en cada momento. Por último, también es una ventaja en todas las funciones del modelo, que cada empresa que intervenga tenga asociada una dirección en la cadena de bloques a la que se puedan enviar pagos por regalías u otros motivos con la criptomoneda Ether.

Otra de las opciones propuestas en el modelo con más potencia desde el punto de vista del ahorro temporal y económico es el cálculo automático de los precios de las licencias. Ciertamente es que, para una situación fuera del Modelo entran en juego infinitas variables para poder evaluar cada caso, pero, esto puede servir como precedente para

desarrollarlo con un mayor grado de complejidad con herramientas más potentes que se escapan del alcance del Proyecto.

En este modelo sigue habiendo una parte centralizada, la cual es difícil de evitar que son los validadores. Se ha visto que ellos mismos son los encargados de, en todas las funciones, verificar que la información es correcta y los procesos son legales. Estos usuarios son designados con esa labor por medio de una entidad centralizada como es la OEPM en España o la USPTO en Estados Unidos. De todas formas, el coste de estos, a simple vista, es inferior al de todos los procesos burocráticos en el sistema actual.

Por último, se considera importante también, destacar la objetividad del modelo que, independientemente de los implicados en el acuerdo, se rige a una serie de parámetros previamente establecidos como pueden ser el tipo de tecnología, la versatilidad de la patente, la duración del acuerdo, el tipo de licencia, la industria de cada implicado o el número de países donde se solicita la licencia. Gracias a esta objetividad, se estaría ahorrando tiempo y dinero, al no tener que especificar el modelo de forma particular para cada situación. De este tipo de ahorros en tiempo y dinero se hablará más a fondo en el próximo capítulo, donde además de sacar conclusiones generales, se analizará el modelo desde el punto de vista del ahorro económico y temporal.

## **5. Memoria económica del Proyecto**

El objetivo del presente capítulo será presentar la inversión inicial que se requeriría para la implementación del Modelo planteado. Para ello, se seguirá la estructura típica propia de una memoria económica. En primer lugar, se presentarán los valores numéricos de los gastos para después explicar el razonamiento detrás de cada uno. Por último, se realizarán unos comentarios al respecto para concluir el análisis económico del proyecto.

### **5.1 Planteamiento de los cálculos**

Es complejo estimar de manera exacta cuánto sería el desembolso económico necesario para realizar lo planteado en el Modelo, aplicar la tecnología blockchain y los contratos inteligentes al sistema de patentes actual. Para llevarlo a cabo se han supuesto unas características del modelo y unas dimensiones que más adelante vendrán detalladas y explicadas.

Es importante destacar también el precio actual de la criptomoneda Ether, que muestra la Figura 43, donde se puede apreciar el valor de la criptomoneda principal de la red de Ethereum en el último año agrupado en velas japonesas de temporalidad una semana, el cual en la actualidad es de 1910 dólares, el precio de esta puede variar con el tiempo, pero los cálculos y las estimaciones en las tasas de la red se realizarán para este valor.



*Figura 43: Gráfica precio actual Ethereum. TradingView, c.2023*

## 5.2 Datos económicos del proyecto

La siguiente Figura 44 muestra los distintos gastos de la implementación del Modelo, estos se han dividido en desarrollo, auditoría y mantenimiento. Cada uno de estos gastos será desglosado y explicado más adelante, detallando el razonamiento detrás de los valores numéricos.

|                                 | Presupuesto   |
|---------------------------------|---------------|
| Desarrollo y puesta en práctica | \$ 105 000.00 |
| Auditoría                       | \$ 25 000.00  |
| Mantenimiento                   | \$ 33 292.80  |
| Gasto total                     | \$ 163 292.80 |

*Figura 44: Datos económicos del proyecto. Elaboración propia, c.2023*

## 5.3 Análisis del Gasto total

### 5.3.1 Aparatos y Equipos físicos y lógicos

El gasto actualmente descrito no se ha incluido en la Figura de los datos económicos del proyecto, porque su valor es 0. Aun así, se quiere destacar que no sería necesario un desembolso económico, ya que con los ordenadores habituales de los que uno dispone en una oficina común son más que suficientes para gestionar e interaccionar dApps que interactúen con los contratos inteligentes.

Con un ordenador de características habituales en 2023, como podría ser un procesador Inter Core i5, una memoria RAM de 8GB y una tarjeta gráfica básica, sería más que suficiente como para utilizar la aplicación. Para una aplicación de las dimensiones de la nuestra, unos 20 000 usuarios y teniendo en cuenta cual sería la función de cada validador, para el correcto funcionamiento y la prevención de la saturación, se estima que harían falta unos 100<sup>60</sup> validadores. Su función se ha venido explicando en capítulos anteriores, pero, esto serían los encargados de validar la información del sistema, trabajarían como “moderadores de la aplicación” y cada uno con su ordenador personal, sería capaz de ejecutar todas las funciones de la aplicación.

### 5.3.2 Costes de Desarrollo

El desarrollo de la aplicación blockchain no podría llevarse a cabo sin el trabajo de desarrolladores expertos en blockchain, concretamente que dominen el lenguaje de programación Solidity. Por ello, se ha decidido que sería necesario contratar a profesionales en la materia para el desarrollo y la auditoría del código detrás de la aplicación blockchain.

El coste del personal dependerá en gran parte de la complejidad de la aplicación blockchain, para este caso, al tratarse de una aplicación para una blockchain empresarial que utiliza indistintamente tokens no fungibles y pagos con criptomonedas, se ha

---

<sup>60</sup> Ethereum. Org c. 2023

considerado de complejidad media. Para esta implementación se contratará a dos<sup>61</sup> desarrolladores blockchain expertos, los cuales aseguren el correcto desarrollo de la aplicación. Esta aplicación tendría también una capacidad de gestión de hasta 20 000<sup>62</sup> usuarios, lo cual encaja con un primer modelo que luego en un futuro pueda escalar y ser capaz de gestionar un mayor número de usuarios. Al tratarse de una aplicación blockchain empresarial, se asegura brindar la máxima seguridad, además de aportar una escalabilidad ilimitada.

| Blockchain App Type as per Complexity | Estimated Blockchain Development Cost | Time Frame    |
|---------------------------------------|---------------------------------------|---------------|
| Simple App                            | \$40,000 to \$60,000                  | 3 to 6 months |
| Moderately Complex App                | \$60,000 to \$150,000                 | 6 to 8 months |
| Highly Complex App                    | \$150,000 to \$300,000                | 9+ months     |

**Figura 45: Estimación de precios y tiempos de desarrollo según la complejidad de la aplicación, Fuente: “A comprehensive Guide on Blockchain App Development Cost”. Sudeep Srivastava (2022)**

Dentro de estos rangos de precios, se estima que alrededor del 95%<sup>63</sup> de la inversión total en el desarrollo, estaría únicamente dedicado a la programación del contrato, donde se incluirían la creación del código y de las pruebas prácticas. Se procederá a desglosar este 95% de forma aproximada, para conocer el coste puro de lanzar un contrato en la red de Ethereum. Habiendo mencionado las características de esta aplicación siguiendo los precios de una aplicación de complejidad media expuestos en la Figura 45, se estima que el coste del desarrollo de la aplicación blockchain sería de unos 105 000 dólares

A continuación, la siguiente Figura 46 desglosará el 95% de los 105 000 dólares necesarios íntegramente para el desarrollo, porque, como se puede apreciar en la siguiente tabla, parte irá también destinado a pruebas en la red.

<sup>61</sup> Sudeep Srivastava, 2022

<sup>62</sup> How much does it cost to get a smart contract made? Iryna Deremuk (2023)

<sup>63</sup> “A comprehensive Guide on Blockchain App Development Cost”. Sudeep Srivastava (2022).

| Sueldos Desarrolladores | Número de Personas | Meses/persona | Horas por semana/persona | Horas Trabajadas/ persona | Precio por hora/persona |
|-------------------------|--------------------|---------------|--------------------------|---------------------------|-------------------------|
| \$ 93 050.00            | 2                  | 7             | 40                       | 1120                      | \$ 41.54                |

| Pruebas en la Red Principal | Número de pruebas | Tarifas de red/prueba | Tarifas de ejecución/prueba | Coste total puesta en práctica/ prueba |
|-----------------------------|-------------------|-----------------------|-----------------------------|--|
| \$ 5 400.00                 | 500               | \$ 3.80               | \$ 7.00                     | \$ 10.80                               |

*Figura 86: Desglose de costes de desarrollo. Elaboración Propia. 2023*

Un contrato inteligente, una vez se ha desarrollado en su totalidad conviene realizar una serie de pruebas para comprobar así el correcto funcionamiento de este en la red de Ethereum. Durante su desarrollo inicial desde 0, se llevan a cabo todas las prácticas en una red de prueba o Testnet, por lo que esto tiene un coste 0. Aún así, conviene probarlo al menos 500<sup>64</sup> veces dentro de la red principal, para comprobar así que está optimizado y no requiere de tarifas de red sin sentido.

El coste medio de una transacción en la red de Ethereum en la actualidad es de aproximadamente 3.8 dólares<sup>65</sup> (0.002 ETH), cuando un usuario ejecuta un contrato inteligente, las tarifas de ejecución son la suma de lo que cuesta ejecutar el contrato inteligente junto con lo que cuesta una transacción en el momento.

El precio asociado a la ejecución de un contrato inteligente dependerá de su nivel de optimización, si asumimos que este es alto debido a que se trata de una aplicación para una blockchain empresarial, el coste de ejecución rondaría los 6-8 dólares<sup>66</sup> (0.0031-0.0042 ETH) para el nivel de saturación actual de la red.

Si se realizan unas 500 pruebas (cantidad recomendada para una aplicación de estas características<sup>67</sup>), el coste asociado a las mismas sería de unos 5 400 dólares, a esto habría que sumarlo el sueldo de dos desarrolladores blockchain, que trabajaran durante 7 meses desarrollando el proyecto. El sueldo de los desarrolladores, como se puede apreciar en la Figura 46, sería de unos 41.54 por hora, haciendo un total de 46 525 dólares en 1120 horas trabajadas cada uno.

<sup>64</sup> MiT Labs, 2022

<sup>65</sup> Ycharts, c.2023

<sup>66</sup> Ycharts, c.2023

<sup>67</sup> Chirag, 2023



Entre las características<sup>68</sup> con las que contaría este desarrollo de aplicación empresarial estarían:

- Interfaces propias, con total libertad de diseño por parte de la empresa encargada, así como con las especificaciones deseadas. Se tendrían que desarrollar dApps dentro del Modelo, que gestionaran las tres funciones del modelo.
- Infraestructura, al no ser desarrollado por las herramientas de código abierto, supondría un mayor desembolso económico de lo habitual. La estructura contaría con sus propios servidores encargados de almacenar su información.
- Red P2P, los pagos dentro de la aplicación se estarían realizando con una seguridad de cifrado impenetrable, también se elegiría el tipo de algoritmo escogido para validar los pagos, en el caso del Modelo sería la Prueba de Participación que utiliza la red de Ethereum.
- La arquitectura del proyecto contaría con una parte centralizada y otra descentralizada, en caso de querer descentralizarlo por completo, la inversión necesaria aumentaría por drásticamente, por lo que habría que valorar si realmente fuera necesario.

Este desarrollo, debido a sus características y sus complejidades, se estima también que tardaría entre 6 y 8 meses<sup>69</sup>, observando el tiempo de desarrollo de proyecto con cualidades similares.

Es importante comentar, que estas estimaciones, tanto de tiempo como de dinero, se han realizado con la premisa de la contratación de una empresa de mediano tamaño<sup>70</sup>, cuya plantilla cuente con al menos 50 expertos en la tecnología blockchain. Al ser un primer modelo de una aplicación que, más tarde sea escalable se considera suficiente con ese personal. En caso de que esta premisa cambiara, habría que valorar de nuevo los costes de desarrollo.

---

<sup>68</sup> MiT Labs, 2022

<sup>69</sup> Chirag, 2023

<sup>70</sup> Chirag, 2023

### 5.3.3 Costes de Auditoria

Por otro lado, una vez el sistema detrás de la aplicación blockchain encargada de gestionar el Modelo esté correctamente programado, habría que hacer frente a unos gastos procedentes de una auditoría de código. Esta parte del proceso es crucial para poder asegurar que los datos de los usuarios estén correctamente protegidos en todo momento<sup>71</sup>. Además, al tratarse en gran parte de los casos de información protegida, es de mayor interés que no pueda darse ningún tipo de fallo en el código permitiendo así filtrar información de forma no deseada.

Siguiendo con las premisas anteriores que se trata de una aplicación blockchain empresarial que, en un primer instante sea capaz de manejar 20 000 usuarios, se estima que el coste asociado a la auditoría del código de los contratos inteligente, donde se describan los posibles problemas y se recomienden mejoras para la seguridad del código en la actualidad y en el futuro, sea de 25 000 dólares.

Este precio se ha justificado con el de otros proyectos con características similares a las del Modelo.



<sup>71</sup> MiT Labs, 2022

*Figura 47: Esquema funcionamiento auditoría SC. MITsoftware. 2022*

La anterior Figura 47, muestra el esquema de funcionamiento de una auditoría de código, donde la empresa encargada, ya sea Certikk, Consensys u otra con gran reconocimiento internacional, aseguraría el correcto funcionamiento, la seguridad y la escalabilidad de la aplicación blockchain para después elaborar un informe.

### **5.3.4 Administración y mantenimiento**

Los actuales gastos irían asociados a revisiones y actualizaciones periódicas del código. Debido al frenético ritmo de desarrollo que lleva la tecnología blockchain, lo recomendable es realizar una auditoría de forma trimestral<sup>72</sup>. Junto con esta, también sería necesario actualizar el contrato inteligente según varíen los protocolos de la red.

En primer lugar, se tendría que llevar a cabo la actualización del código, esto, podríamos suponer que se realizara cada tres meses, con una duración de 1 semana e interviniendo dos profesionales en el proceso<sup>73</sup>. Después, se llevaría a cabo la auditoría del código para validar el correcto funcionamiento de este, así como la seguridad de los usuarios que lo utilizan. Debido a que es esta auditoría sería de mantenimiento y actualización el precio se estima que sería menor<sup>74</sup> que el de verificación de un código programado desde 0.

La siguiente Figura 48 muestra los costes anuales asociados a esta parte de la implementación:

---

<sup>72</sup> MIT Labs, 2022

<sup>73</sup> MIT Labs, 2022

<sup>74</sup> Binance, 2022

| Costes de actualización    | Número de Personas                               | Horas Trabajadas/ persona                   | Precio por hora/persona |
|----------------------------|--|---|-------------------------|
| \$ 3 323.20                | 2  | 40  | \$ 41.54                |
| <b>Costes de auditoría</b> |  |   |                         |
| \$ 5 000.00                |  |   |                         |
| <b>Coste total anual</b>   | <b>Número de actualizaciones necesarias/ año</b> | <b>Número de auditorías necesarias/ año</b> |                         |
| \$ 33 292.80               | 4  | 4   |                         |

*Figura 48: Costes de actualización y auditoría anuales desglosados. Elaboración Propia, c.2023*

En un futuro, si el modelo escala, estas revisiones trimestrales no serían necesarias, ya que se aspira a contar con personal especializado en la tecnología blockchain contratado de manera interna.

## 5.4 Conclusiones

Para concluir, dejar claro que, a la hora de realizar los cálculos y estimaciones se ha tratado de ser lo más exacto posible basándose en proyectos con características similares al nuestro o mediante datos obtenidos directamente al describir las cualidades de nuestra aplicación blockchain.

Cuando se ha representado el coste total al principio del capítulo, se ha incluido el mantenimiento del primer año, pero después se explica que cada una de esas actualizaciones y auditorías se realizarían de manera trimestral. También se ha dejado claro que estos costes son para una aplicación de unos 20 000 usuarios activos, pero, en un futuro se podría aumentar la capacidad de gestión de la misma. En un futuro, se buscaría tener personal contratado de manera interna dentro de las oficinas de patentes a nivel mundial que sean capaces de desarrollar y asegurar contratos inteligentes.

Las primeras impresiones pueden ser que la implementación sea costosa y no merezca la pena invertir tales cantidades de dinero a cambio de transformar el funcionamiento

del sistema tradicional de patentes. Todas esas transformaciones van de la mano de incrementos en la seguridad, automatización y ahorro en el tiempo por parte de los usuarios.

Por último, dejar claro que a medida que la adopción de la tecnología blockchain vaya aumentando, probablemente los costes de elaboración de aplicaciones blockchain decrezcan al ser más accesibles. Por lo tanto, el coste del primer año de implementación del modelo, 163 292 dólares representan una cantidad monetaria pequeña frente al gran cambio que la implementación de esta tecnología puede suponer. Una vez se ha realizado la inversión inicial, los costes de mantenimiento de la aplicación con el avance de las tecnologías podrían llegar a verse reducidos si se optimiza la red de Ethereum o se puede automatizar los procesos de desarrollo de código.

En el próximo capítulo se expondrán unas conclusiones generales del Proyecto, tratando de resaltar y sintetizar las ideas y conceptos recogidos a lo largo del Modelo.

## 6. Conclusiones

En el presente capítulo se expondrán los objetivos conseguidos en el Proyecto, así como las ideas que se han considerado importantes recogidas durante el desarrollo de este. Además, al final, se incluirán unos breves comentarios enfocados a futuros desarrollos del Modelo actual con intención de mejorarlo.

### 6.1 Conclusiones y Objetivos

En el presente Proyecto se ha elaborado un Modelo el cual anexiona la tecnología blockchain al Sistema de Patentes Tradicional e incorpora las características de esta. Se ha conseguido plantear un modelo que cuente con la seguridad, transparencia, escalabilidad y efectividad de la tecnología de la blockchain. Por medio del Modelo planteado basado en los nodos de la red de Ethereum y los validadores propios de esta aplicación blockchain, se ha hecho posible el funcionamiento de tres gestiones distintas relacionadas con patentes de manera automatizada gracias al código programado en Solidity.

El Modelo se ha conseguido plantear por medio de diagramas de flujos (de forma teórica) y mediante variables de entrada y salida, que, en la mayoría de los casos, se trataba de un tensor (de forma matemática). En estas variables se ha conseguido almacenar la información referente a las patentes que, tras un proceso de recopilación informativo en los primeros capítulos, se ha considerado importante.

Además, se ha logrado plantear de forma completa un proceso de verificación y concesión del acceso a determinada información privada por medio de activos criptográficos conocidos como NFTs que servirían como certificado digital, concediendo el acceso al usuario. Estos NFTs también tendrían una función de almacenamiento de información. Dentro de sus metadatos estaría incluida toda la información referente a una patente y estaría almacenado de forma privada, queriendo decir esto que solo la dirección en la cadena de bloques dueña de ese NFT podría acceder a sus metadatos y visualizar su información.

También, en el Proyecto se ha conseguido plantear un modelo formado por tres funciones; Creación de nuevas patentes, Solicitar licencias de explotación y Gestionar acuerdos de licencias cruzadas. Estas funciones cuales automatizan procesos por medio de la programación de un contrato inteligente en Solidity Por otro lado, incluido en cada una de las funciones, se ha conseguido formular un algoritmo capaz de calcular de forma objetiva el precio de cada patente adaptándose a cada situación en base a unos parámetros.

Esto último se ha logrado gracias a la elaboración de un cuestionario que sirve para categorizar cada licencia y proporcionar la información necesaria a la calculadora de precios de patentes desarrollada en el Proyecto. Este cuestionario lo rellenaría el licenciante una vez se le ha concedido la patente y esta se introduce en el sistema del Modelo. Por lo tanto, como se ha observado, esta parte del modelo habría logrado establecer el precio de cada patente de manera objetiva en base a distintos parámetros detallados en el Capítulo 3(duración del acuerdo, territorialidad del acuerdo, exclusividad del acuerdo, etc.), gracias a esto, nos evitaríamos negociaciones que consuman tiempo y dinero por ambas partes.

También se ha elaborado una Memoria Económica la cual representa la inversión inicial necesaria para llevar a cabo la implementación del proyecto. A la hora de evaluar los gastos, estos se han estudiado de manera realista apoyándonos en datos proporcionados por aplicaciones blockchain con características similares. Los costes totales de financiación del Proyecto, para el primer año, teniendo en cuenta la inversión inicial en el desarrollo de la aplicación desde 0 y el mantenimiento del primer año (actualizaciones y revisiones realizadas trimestralmente) ascendería a los 163 292 dólares.

Con el funcionamiento del Modelo se estaría consiguiendo automatizar y ahorrar en tiempo y dinero en procesos puramente burocráticos por medio de la programación del código del contrato inteligente. Habría que añadirle a este ahorro económico la mejora en la seguridad gracias a la criptografía de la tecnología blockchain.

Por otro lado, todas estas gestiones contarían con la transparencia y trazabilidad propia de la cadena de bloques, por lo tanto, se estarían llevando a cabo las gestiones que

cubren nuestras tres funciones (creación de una nueva patente, gestión de acuerdos de licencias cruzadas y solicitud de licencias de explotación) de forma totalmente transparente y segura.

Aunque esta inversión pudiera parecer elevada, al tratarse de nuevas tecnologías, como se ha podido observar a lo largo del Proyecto, se estaría abriendo la puerta a la innovación y el desarrollo, incorporando una tecnología con un gran potencial como es la tecnología blockchain. Por ello, convendría ponerlo en práctica con la capacidad de usuarios actual (20 000) para después sacar futuras conclusiones y valorar posibles mejoras y optimizaciones.



## 6.2 Desarrollos futuros

La tecnología blockchain se encuentra en constante desarrollo y, por ello mismo, se ha considerado oportuno desarrollar esta parte dentro del actual capítulo dirigido más hacia acciones y recomendaciones futuras que aseguren la viabilidad y evolución de este a lo largo del tiempo.

Ha quedado claro en anteriores capítulos que el Modelo propuesto funcionaría como una aplicación base, y que, el objetivo en un futuro sería desarrollarlo. Una de las características más potentes de la tecnología blockchain es la escalabilidad que esta posee, facilitando el aumento del número de usuarios por medio de la programación de un nuevo código.

Una de las mejoras con las que podrían contar futuras versiones del Modelo, podría ser la incorporación de un algoritmo encargado de, cuando el usuario va a solicitar una patente, avisar en caso de que el acuerdo de licencias cruzadas fuera de interés mutuo. Es decir, si la empresa A fuese a solicitar una patente a la empresa B, en caso de que la empresa B estuviera previamente utilizando una patente de la empresa A, esto se le notificara a la empresa A antes de solicitar la patente a la empresa B. Esto, facilitaría las gestiones a los usuarios, al ahorrar tiempo de investigación referente a su situación actual frente a sus futuros licenciantes.

Otra de las mejoras del Modelo para futuras versiones, sería tratar de desarrollar al máximo el algoritmo que calcula el precio de las patentes. Si este llegase a ser lo suficientemente complejo, podría servir unificarse y ser utilizado por las distintas oficinas de patentes, significando un paso hacia delante en la unión de todas las oficinas de patentes mundiales y promoviendo la innovación y el desarrollo al simplificar los procesos mundiales de invención.

Para agilizar todavía más esos cambios en las características del Modelo, sería interesante implementar el uso de inteligencia artificial para la programación de este nuevo código. De esta forma se estaría ahorrando tiempo y dinero invertido en programación al estar delegándolo en una IA capaz de programar contratos inteligentes que cuenten con las nuevas características del momento.

Por otro lado, en un futuro también se buscaría tratar de descentralizar al máximo el modelo, pero para ello, primero habría que ver cuál es la respuesta del sistema de patentes actual a nuestro modelo. Para ello habría que desarrollar más todavía la función de los nodos validadores, así como tratar de unificar de manera global las oficinas de patentes mundiales.

Este objetivo hoy en día suena inalcanzable, pero con el constante avance de las tecnologías y la globalización actual, en un futuro, en caso de funcionar modelos similares al planteado, podría llegar a suceder, siempre y cuando la tecnología blockchain continuara por la senda de la adopción y el desarrollo.

## Bibliografía

### Libros y Artículos

#### Smart contracts

- Antonopoulos, Andreas, “Mastering Ethereum: Building Smart contracts and DApps”, O’Reilly Media, ISBN: 9781491971895, 2018.
- Appel, Gil; Malik, Nikhil; Luo, Lan; Wei, Yanhao, “Blockchain technology for creative industries: Current state and research opportunities”, International Journal of Research in Marketing, Vol 40, 38-48, 2022.
- Bamakan, Hosseini; Bodaghi, Omid; Mojtaba, Seyed; Nezhadsistani, Nasim; Qu, Qiang, ”Patents and intellectual property assets as non-fungible tokens; key technologies and challenges”, Nature: Scientific Reports, 2022.
- Cervantes Maceda,Humberto; O. Solís Osorio, Carlos; Pérez Cortés, Elizabeth, “Hacia una metodología para el diseño de contratos inteligentes”, Redalyc, Vol 8, 1-15, 2019.
- Chirag, “A Digestible Guide to Blockchain App Development”, AppInventiv, 2023.
- Deremuk, Iryna, “How much does it cost to deploy a Smart Contract?”, Litslink, 2022.
- Huang, Yuxin, “Research and Application of Smart Contract Based on Ethereum Blockchain”,Journal of Physics Conference Series, Vol 4,1-13, 2021.
- Ilaria Lunesu, María; Marchesi, Lodovica; Marchesi, Michele; Tonelli, Roberto, “A blockchain architecture for industrial applications”, El Sevier, Vol 3, 1-13, 2022.
- Lesnova, N.I.; Mukhtarova, A.R., “Smart contracts in international trade in services in the field of international property”, Atlantis Press, Vol 105, 523-527, 2018.
- MiT Labs, “Smart Contract Audit Certification Service”, MiT Software, 2022.

### **Patentes y licencias cruzadas**

- Giuri, Paola; Torrisi, Salvatore, “Cross-licensing, Cumulative Inventions and Strategic Patenting”, Department of Management, University of Bologna, Vol 1, 2-29, 2010.
- Jeon, Don-Shin; Lefouili, Yassine, “Cross-licensing and competition”, JSTOR, Vol 49, 656-671, 2018.
- Rose, Anne, “Transforming the registration of IP rights and strengthening the protection of unregistered IP rights”, WIPO, 2020.
- Sánchez García, Luz, “EL FENÓMENO CROSS-LICENSING EN EL DERECHO ESPAÑOL DE PATENTES”, Revista de Direito Brasileira, Vol 19, 141-154, 2018.
- Sidak, J. Gregory, “En qué se parece la negociación de licencias de patentes esenciales a comprar un automóvil nuevo”, WIPO, 2015.
- Shapiro, Carl, “Navigating the Patent Thicket: Cross Licenses, Patent Pools, and Standard Setting”, Mit Press, Vol 1, 119-150, 2001.

### **Licencias de Patentes**

- Cahill, M. Jason; Hopmann; Alexander, Keith Windle, Marc; Raymundo Lerma, Erick, “Server Farm Management”, 2015.
- Chen, Xiaogang; Li, Qinghua; Zhu, Yuan, “Mapping and enhanced physical downlink control channel”, 2017.
- Chmielewski, Harry; Ducker, Paul; Vrooman, Jacob; Kaiser, Thomas; Sergeant, Tim, “Artículo Absorbente con núcleo absorbente plegado de capa múltiple”, 2015.
- J.Pande, Vinay; Biswas, Kaushik; Chokshi, Jayesh; G.Wing, Daniel, “Intelligent ALG functionality in networking supporting endpoints performing network address translation”, 2006.
- Knecht, Theresa, “Procedimiento para producir pañales desechables para incontinencia”, 2017.

- Lim, Pyung-Bin, “Integrated Internet telephony system and signaling method thereof”, 2006.
- Tedaldi, David, Mermoud, Grégory; Savalle, Pierre-Andre; Vasseur, Jean-Philippe, “Learning Robust and accurate rules for device classification from clusters of devices”, 2019.

### **Páginas web**

Oficina Española de patentes y marcas

- [https://www.oepm.es/es/propiedad\\_industrial/propiedad\\_industrial/que\\_se\\_pued\\_e\\_proteger\\_y\\_como/](https://www.oepm.es/es/propiedad_industrial/propiedad_industrial/que_se_pued_e_proteger_y_como/)
- [http://www.oepm.es/es/propiedad\\_industrial/tasas/](http://www.oepm.es/es/propiedad_industrial/tasas/)
- [https://www.oepm.es/es/propiedad\\_industrial/internacionalizacion/patentes\\_modelos\\_utilidad/index.html](https://www.oepm.es/es/propiedad_industrial/internacionalizacion/patentes_modelos_utilidad/index.html)
- [https://www.oepm.es/es/sobre\\_oepm/noticias/2020/2020\\_01\\_24\\_EvolucionSolicitudesNacionalesPI\\_2019\\_provis.html](https://www.oepm.es/es/sobre_oepm/noticias/2020/2020_01_24_EvolucionSolicitudesNacionalesPI_2019_provis.html)

Organización Mundial de Patentes y Marcas

- <https://www.wipo.int/portal/es/>

Ethereum

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

Nchain, empresa que actualmente basa su funcionamiento en smart contracts.

- <https://nchain.com/company/about-nchain/>

Cointelegraph

- <https://cointelegraph.com/ethereum-for-beginners/what-are-smart-contracts-a-beginners-guide-to-automated-agreements>
- <https://cointelegraph.com/ethereum-for-beginners/what-is-ethereum-a-beginners-guide-to-eth->

[cryptocurrency#:~:text=Ethereum%20is%20a%20decentralized%20blockchain,media%20and%20so%20much%20more.](#)

- <https://cointelegraph.com/news/vyper-solidity-and-scrypto-how-the-smart-contract-languages-compare>
- <https://cointelegraph.com/news/what-is-an-ethereum-virtual-machine-evm-and-how-does-it-work>

#### Google Patents

- <https://patents.google.com/>

#### Programación smart contracts

- <https://openai.com/blog/chatgpt>

#### Precio de transacción en la red de Ethereum

- [https://ycharts.com/indicators/ethereum\\_average\\_transaction\\_fee](https://ycharts.com/indicators/ethereum_average_transaction_fee)

## Anexo A

Código Solicitar Licencia de Explotación

```
pragma solidity ^0.8.0;
```

```
contract Patents {
```

```
    uint256 constant maxLicenseNumber = 9999999;
```

```
    struct Patent {
```

```
        bool available;
```

```
        uint256 price;
```

```
        address licenciate;
```

```
    }
```

```
    mapping (uint256 => Patent) public patents;
```

```
    mapping (uint256 => mapping(address => bool)) public allowedUsers;
```

```
    event PatentRegistered(uint256 licenseNumber, uint256 price, address licenciate);
```

```
    event AccessGranted(uint256 licenseNumber, address user);
```

```
    event PaymentReceived(address payer, uint256 amount);
```

```
    event AgreementBroken(address payer, uint256 amount);
```

```
    function registerPatent(uint256 licenseNumber, uint256 price) public {
```

```
        require(licenseNumber <= maxLicenseNumber, "Invalid license number");
```

```
        require(price > 0, "Price must be greater than 0");
```

```
        require(!patents[licenseNumber].available, "Patent already registered");
```

```
        patents[licenseNumber] = Patent(true, price, msg.sender);
```

```
emit PatentRegistered(licenseNumber, price, msg.sender);
}

function negotiatePatent(uint256 licenseNumber, uint256 price) public payable {
    require(licenseNumber <= maxLicenseNumber, "Invalid license number");
    require(msg.value == price, "Incorrect price");
    require(patents[licenseNumber].available, "Patent not available for negotiation");

    patents[licenseNumber].available = false;
    patents[licenseNumber].price = price;

    emit PaymentReceived(msg.sender, msg.value);
}

function allowAccess(uint256 licenseNumber) public payable {
    require(licenseNumber <= maxLicenseNumber, "Invalid license number");
    require(patents[licenseNumber].licenciate == msg.sender, "Access denied, only the
licenciate can grant access");
    require(patents[licenseNumber].available == false, "Access denied, patent not sold
yet");
    require(patents[licenseNumber].price == msg.value, "Incorrect payment amount");

    allowedUsers[licenseNumber][msg.sender] = true;

    emit AccessGranted(licenseNumber, msg.sender);
}

function checkAccess(uint256 licenseNumber, bool userNegotiated) public view
returns (bool) {
    require(licenseNumber <= maxLicenseNumber, "Invalid license number");
```



```
    if (userNegotiated) {
        return allowedUsers[licenseNumber][msg.sender];
    } else {
        return patents[licenseNumber].available == false &&
allowedUsers[licenseNumber][msg.sender];
    }
}

function breakAgreement(uint256 licenseNumber) public {
    require(licenseNumber <= maxLicenseNumber, "Invalid license number");
    require(!patents[licenseNumber].available, "Patent not sold yet");
    require(allowedUsers[licenseNumber][msg.sender], "Access denied");

    patents[licenseNumber].licenciate.transfer(patents[licenseNumber].price);
    allowedUsers[licenseNumber][msg.sender] = false;

    emit AgreementBroken(msg.sender, patents[licenseNumber].price);
}
}
```

## Anexo B

Código Acuerdo de Licencias Cruzadas

```
pragma solidity ^0.8.0;
```

```
contract LicensesCross {
```

```
    // Variables de entrada
```

```
    uint[] public patentNumbers;
```

```
    uint public company1Id;
```

```
    uint public company2Id;
```

```
    string public conditions;
```

```
    bool public started;
```

```
    // Dirección de la empresa 2
```

```
    address public company2Address;
```

```
    // Patentes ofrecidas por cada empresa
```

```
    mapping(uint => bool) public company1OfferedPatents;
```

```
    mapping(uint => bool) public company2OfferedPatents;
```

```
    // Variables para el pago en ether
```

```
    bool public paymentRequired;
```

```
    uint public paymentAmount;
```

```
    // Estados de la negociación
```

```
    enum AgreementState { Pending, Approved, Rejected }
```

```
    AgreementState public agreementState;
```

```
    address public company1Signer;
```

```
    address public company2Signer;
```

```
    // Eventos
```

```
event Company2ContactInfo(address company2Address);
event Company2OfferedPatents(uint[] patentNumbers);
event AgreementUpdated(AgreementState state);

// Función para establecer las variables de entrada
function setAgreementData(
    uint[] memory _patentNumbers,
    uint _company1Id,
    uint _company2Id,
    string memory _conditions,
    bool _started
) public {
    require(_patentNumbers.length == 5, "Debe proporcionar 5 números de patentes");
    require(bytes(_conditions).length > 0, "Debe proporcionar las condiciones");

    patentNumbers = _patentNumbers;
    company1Id = _company1Id;
    company2Id = _company2Id;
    conditions = _conditions;
    started = _started;

    // Comprobar que los números de patentes y los números identificativos de las
    // empresas son válidos (no se implementará la verificación)
}

// Función para que la empresa 1 obtenga la información de contacto de la empresa 2
function getCompany2ContactInfo() public {
    require(started == false, "La negociación ya ha empezado");

    // Obtener la dirección de la empresa 2 (no se implementará la obtención)
```

```
company2Address =
address(0x1234567890123456789012345678901234567890);

emit Company2ContactInfo(company2Address);
}

// Función para que la empresa 2 ofrezca las patentes que le interesan de la empresa 1
function offerPatents(uint[] memory _patentNumbers) public {
    require(msg.sender == company2Address, "Solo la empresa 2 puede hacer una oferta");
    require(_patentNumbers.length > 0, "Debe proporcionar al menos una patente");

    // Añadir las patentes ofrecidas por la empresa 2
    for (uint i = 0; i < _patentNumbers.length; i++) {
        company2OfferedPatents[_patentNumbers[i]] = true;
    }

    emit Company2OfferedPatents(_patentNumbers);
}

// Función para que la empresa 2 acepte o modifique las condiciones de la empresa 1
function acceptConditions(string memory _newConditions) public {
    require(msg.sender == company2Address, "Solo la empresa 2 puede aceptar o modificar las condiciones");

    // Si se proporcionan nuevas condiciones, se actualizan
    if (bytes(_newConditions).length > 0) {
        conditions = _newConditions;
    }

    // El pago no es requerido en este punto
    paymentRequired = false;
}
```

```
// Actualizar el estado de la negociación a Aprobada
agreementState = AgreementState.Approved;
company2Signer = msg.sender;

emit AgreementUpdated(agreementState);
}

// Función para que la empresa 2 rechace las condiciones de la empresa 1
function rejectConditions() public {
    require(msg.sender == company2Address, "Solo la empresa 2 puede rechazar las
condiciones");

    // El pago no es requerido en este punto
    paymentRequired = false;

    // Actualizar el estado de la negociación a Rechazada
    agreementState = AgreementState.Rejected;

    emit AgreementUpdated(agreementState);
}

// Función para que la empresa 1 inicie el proceso de pago
function startPayment(uint _paymentAmount) public {
    require(agreementState == AgreementState.Approved, "La negociación no ha sido
aprobada");
    require(msg.sender != company2Address, "Solo la empresa 1 puede iniciar el proceso
de pago");

    paymentRequired = true;
    paymentAmount = _paymentAmount;
```

```
}
```

```
// Función para que la empresa 2 confirme el pago y finalice la negociación
```

```
function confirmPayment() public {
```

```
    require(msg.sender == company2Address, "Solo la empresa 2 puede confirmar el  
pago");
```

```
    require(paymentRequired == true, "El pago no es requerido en este momento");
```

```
    // Finalizar la negociación
```

```
    agreementState = AgreementState.Pending;
```

```
    paymentRequired = false;
```

```
    emit AgreementUpdated(agreementState);
```

```
}
```

```
}
```

## Anexo C

Código Crear Nueva Patente

```
pragma solidity ^0.8.0;
```

```
contract PatentSystem {
```

```
    struct Patent {
```

```
        string patentInfo;
```

```
        string stateOfTheArt;
```

```
        address owner;
```

```
        uint256 date;
```

```
        bool isLicensed;
```

```
        address[] validators;
```

```
    }
```

```
    Patent[] public patents;
```

```
    mapping(address => bool) public validators;
```

```
    modifier onlyValidator() {
```

```
        require(validators[msg.sender], "You are not authorized to access this function.");
```

```
        _;
```

```
    }
```

```
    modifier onlyPatentOwner(uint256 _id) {
```

```
        require(patents[_id].owner == msg.sender, "You are not the owner of this patent.");
```

```
        _;
```

```
    }
```

```
    function addValidator(address _validator) public {
```

```
    validators[_validator] = true;
}

function removeValidator(address _validator) public {
    validators[_validator] = false;
}

function createPatent(string memory _patentInfo, string memory _stateOfTheArt)
public {
    Patent memory newPatent = Patent({
        patentInfo: _patentInfo,
        stateOfTheArt: _stateOfTheArt,
        owner: msg.sender,
        date: block.timestamp,
        isLicensed: false,
        validators: new address[](0)
    });

    patents.push(newPatent);
}

function getPatentInfo(uint256 _id) public view onlyValidator() returns (string
memory) {
    return patents[_id].patentInfo;
}

function getStateOfTheArt(uint256 _id) public view onlyValidator() returns (string
memory) {
    return patents[_id].stateOfTheArt;
}
```



```
function getPatentDate(uint256 _id) public view onlyValidator() returns (uint256) {
    return patents[_id].date;
}

function addValidatorToPatent(uint256 _id, address _validator) public
onlyPatentOwner(_id) {
    patents[_id].validators.push(_validator);
}

function removeValidatorFromPatent(uint256 _id, address _validator) public
onlyPatentOwner(_id) {
    for (uint256 i = 0; i < patents[_id].validators.length; i++) {
        if (patents[_id].validators[i] == _validator) {
            delete patents[_id].validators[i];
            break;
        }
    }
}

function grantLicense(uint256 _id) public onlyValidator() {
    require(!patents[_id].isLicensed, "This patent has already been licensed.");

    patents[_id].isLicensed = true;
}
}
```