Estudio y evaluación de la identidad digital en Blockchain

Trabajo Fin de Máster, junio del 2020

Máster Interuniversitario en Seguridad de las T.I.C.

Autor:

Denis Ionut Stefanescu

Tutores:

Enric Hernández Jiménez (UOC)

Xabier Larrucea Uriarte (Tecnalia)

Responsable área:

Víctor García Font (UOC)


Universitat
Oberta
de Catalunya


Universitat Autònoma
de Barcelona


UNIVERSITAT
ROVIRA I VIRGILI


UIB
Universitat de les
Illes Balears

RESUMEN

Este proyecto consiste en un estudio teórico de la identidad descentralizada basada en Blockchain (identidad soberana), así como las soluciones existentes para la gestión de la identidad en Blockchain. Dichas soluciones son: Alastria ID, Sovrin, Hyperledger Indy y uPort. Finalmente, también se presenta una pequeña tabla comparativa de las principales características de cada una de las soluciones que se han analizado.

La segunda parte de este proyecto (parte práctica) consiste en diseñar y poner en marcha una solución de identidad soberana partiendo de la base de alguna de las soluciones examinadas en el apartado teórico (Hyperledger Indy).

Posteriormente se realizará una adaptación del sistema que se ha puesto en marcha para el ámbito de la salud, concretamente para el traspaso de expedientes médicos entre diferentes instituciones sanitarias de diferentes comunidades autónomas o países, teniendo en cuenta el actual Reglamento de Protección de Datos (GDPR) de la Unión Europea (UE).

Finalmente, para cumplir el objetivo principal del proyecto, se hará una evaluación del sistema de intercambio de expedientes puesto en marcha con Hyperledger Indy.

Palabras clave: Blockchain, identidad digital, identidad soberana, Hyperledger Indy, healthcare.

ABSTRACT

This project consists of a theoretical study of decentralized identity (blockchain - sovereign identity), as well as the existing solutions for decentralized identity management. These solutions are: Alastria ID, Sovrin, Hyperledger Indy and uPort. Finally, a comparative table of the main characteristics of each of the solutions that have been analyzed is also presented.

The second part of this project consists of designing and implementing a sovereign identity solution based on one of the solutions examined in the theoretical section (Hyperledger Indy).

Subsequently, an adaptation of the system that has been implemented for the healthcare field will be carried out, specifically for the transfer of medical records between different healthcare institutions in different regions or countries, taking into account the current Data Protection Regulation (GDPR) of the European Union (EU).

Finally, in order to fulfil the main objective of the project, an evaluation of the exchange system implemented with Hyperledger Indy will also be carried out.

Keywords: Blockchain, digital identity, self-sovereign identity, Hyperledger Indy, healthcare.

ÍNDICE

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

1. Introducción

1.1 Contexto y justificación

La utilización de Distributed Ledger Technologies (DLT) está demostrando una serie de ventajas para la sociedad y para las empresas debido a las funcionalidades de trazabilidad que ofrecen, entre otras. Su aplicación ha sido demostrada en varios dominios y entornos. En concreto, se ha planteado la utilización de DLTs para el intercambio de expedientes (registros) médicos entre diferentes países europeos. Sin embargo, surgen algunas dudas cuando se aplican a dominios como la salud y el rol de la identidad en este tipo de entornos DLTs. [59, 60] Por lo tanto, resulta de vital importancia investigar y desarrollar mecanismos que ayuden a la gestión de la identidad digital (identidad soberana) en este tipo de entornos. [59]

Todo esto surge a raíz del importante crecimiento y protagonismo de la tecnología Blockchain en diferentes ámbitos, siendo uno de ellos la gestión de la identidad digital de manera que el usuario tenga un control pleno sobre la gestión de sus datos. Este nuevo concepto revolucionario ha captado la atención de algunas organizaciones, individuos y empresas. Todos ellos preocupados por la privacidad y el acceso a sus datos en un mundo interconectado como lo es el mundo actual.

Durante el desarrollo del trabajo se realizará un estudio más exhaustivo sobre el estado del arte de la identidad digital soberana en el ámbito de la medicina. Dicho estudio servirá como apartado introductorio del capítulo enfocado a la puesta en marcha de un sistema de identidad digital soberana para el intercambio de expedientes médicos, siendo este capítulo el objetivo principal de este proyecto, tal y como se detalla en el apartado siguiente, 1.2 “Objetivos del proyecto”.

1.2 Objetivos del proyecto

El objetivo principal del proyecto es estudiar y evaluar la identidad digital (soberana) en Blockchain (DLT) en un entorno como es el intercambio de expedientes médicos entre distintos países europeos (teniendo en cuenta GDPR).

Por lo tanto, para llegar a conseguir el objetivo planteado, se han definido una serie de subobjetivos o tareas:

1. Realizar la planificación del proyecto.
2. Redactar un capítulo para introducir los conceptos básicos relacionados con este trabajo: Blockchain, identidad digital e identidad soberana.
3. Análisis de las soluciones existentes para la gestión de la identidad en Blockchain.
4. Diseño y desarrollo de un método o aplicación que asegure la identidad o proporcione garantía sobre ella (sistema de identidad soberana).
5. Puesta en marcha y pruebas sobre el sistema desarrollado.
6. Análisis de la identidad soberana en el ámbito concreto de la medicina.
7. Puesta en marcha de un DLT para el intercambio de expedientes médicos.
8. Redactar la documentación y realizar la presentación del proyecto.

1.3 Metodología

El enfoque y la metodología a seguir para cumplir con los objetivos marcados en el presente trabajo vienen definidos por las siguientes etapas. Cada etapa contará con un capítulo propio dentro de la memoria final, por lo que este apartado también equivaldría a una descripción de la estructura de la memoria del trabajo.

Planificación del trabajo

Esta primera etapa cosiste en identificar el contexto y justificar la importancia de realizar un proyecto en el área de Blockchain aplicado a la gestión de la identidad. Además, se definen los objetivos del trabajo, la metodología utilizada, las tareas necesarias para cumplir los objetivos, se identifican los principales riesgos, se realiza una planificación temporal de las tareas y una estimación de costes.

Análisis de los conceptos previos básicos

Esta segunda etapa consiste en una introducción a los conceptos básicos previos relacionados con la temática del proyecto: Blockchain, identidad digital e identidad soberana.

Parte teórica: estudio de las soluciones existentes

La tercera etapa es el desarrollo de la parte teórica del proyecto. Se analizarán algunas soluciones fundamentales dentro del ámbito de la identidad digital descentralizada. Dichas soluciones son: Alastria (Alastria ID), Sovrin, Hyperledger (Hyperledger Indy) y uPort. Finalmente, también se presentará una tabla comparativa de las soluciones analizadas en donde se podrán visualizar las diferencias entre algunos aspectos técnicos de las mismas.

Diseño, desarrollo y testeo de un sistema de identidad soberana

La cuarta etapa consiste en el diseño, puesta en marcha y testeo de un sistema de identidad soberana compuesto por tres actores (dos organizaciones y una persona física). El sistema estará basado en una de las soluciones analizadas en la etapa del desarrollo teórico. En esta etapa se definirá la base que llevará al cumplimiento del objetivo principal del proyecto, es decir, la puesta en marcha de un sistema de identidad digital soberana para el intercambio de información médica entre distintas regiones o países teniendo en cuenta la actual normativa de protección de datos de la Unión Europea (GDPR).

Puesta en marcha de un sistema de identidad soberana en el ámbito de la medicina

La quinta etapa se compondría de 3 partes:

- Una pequeña introducción sobre la temática del intercambio de expedientes médicos, exponiendo los problemas actuales y la necesidad de investigar y desarrollar nuevas soluciones innovadoras para dicha temática. Toda la información presentada en esta primera parte estará basada en distintos artículos científicos que tratan sobre este tema.
- La modificación del sistema desarrollado en la etapa anterior para la temática del intercambio de los expedientes médicos y su puesta en marcha junto a una demostración empleando un caso de uso específico.
- La evaluación de la identidad digital en sistema que se ha puesto en marcha.

Informe de recursos destinados al proyecto

La sexta y penúltima etapa, “informe de recursos destinados al proyecto”, trata sobre la dedicación temporal del trabajo. Se van a comparar los tiempos estimados y los tiempos reales que se han empleado en el desarrollo del TFM.

Conclusiones y trabajo futuro

Finalmente, en la última etapa del trabajo se expondrán las conclusiones a las que se ha llegado tras realizar el proyecto, así como las líneas futuras y las mejoras pendientes.

2. Visión global del proyecto

La metodología ágil empleada en este proyecto produce resultados parciales por cada iteración en la adquisición de requisitos, en las tomas de decisiones y los desarrollos encaminados a alcanzar los objetivos definidos en cada iteración.

Sin embargo, para una mayor legibilidad y almacenamiento de la documentación, en este capítulo se muestra la visión global inicial definida al inicio del proyecto.

2.1 Análisis de requisitos

2.1.1 Requisitos funcionales iniciales

Realizar un estudio teórico sobre los siguientes conceptos: Blockchain, identidad digital, identidad soberana y las soluciones existentes para la gestión de la identidad en Blockchain. También se estudiará el concepto de la identidad digital en Blockchain aplicada a un área específico como el intercambio de expedientes médicos entre distintos países teniendo en cuenta aspectos legales como el Reglamento General de Protección de Datos (GDPR) vigente en la Unión Europea.

Diseñar y desarrollar un método o aplicación que asegure la identidad o proporcione garantía sobre ella, es decir, un sistema de identidad soberana.

Finalmente, se requiere poner en marcha un Distributed Ledger Technology (DLT) para el intercambio de expedientes médicos.

2.1.2 Tecnología, equipo y conceptos a desarrollar

Tecnología

En este apartado se van a comentar las diferentes tecnologías empleadas en el desarrollo de este proyecto.

- Ordenador de sobremesa personal (PC) con Microsoft Windows 10.
- Microsoft Office 2016 para la documentación del proyecto.
- Hyperledger Indy

Equipo del proyecto

Alumno, encargado de realizar el proyecto:

- Denis Ionut Stefanescu

Directores y profesores colaboradores, encargados de hacer un seguimiento del proyecto y de brindar apoyo técnico y/o teórico:

- **Tutor 1:** Enric Hernández Jiménez (UOC)
- **Tutor 2:** Xabier Larrucea Uriarte (Tecnalia)
- **Responsable del área:** Víctor García Font (UOC)

Conceptos a desarrollar

- Blockchain aplicado a la gestión de la identidad digital (identidad soberana)
- Diseño y desarrollo con Hyperledger Indy
- La identidad soberana en el ámbito de la salud: intercambio de expedientes médicos

3. Planificación

3.1 Introducción

Para explicar la planificación que se ha seguido para el desarrollo del trabajo, primero se presentará la estimación de los tiempos, en los cuales se prevé que se va a desarrollar adecuadamente el trabajo a realizar. Después se explicarán las fases de desarrollo que se han seguido para la realización del proyecto y para finalizar, una estimación de costes del proyecto.

3.2 Diagrama de Gantt

Un diagrama de Gantt es una herramienta que sirve para la planificación de proyectos. Proporciona una vista general de las tareas programadas y ayuda a saber qué tareas se deben realizar y en qué fechas.

En el caso de este proyecto se han definido 8 tareas principales:

1. Realizar la planificación del proyecto.
2. Redactar un capítulo para introducir los conceptos básicos relacionados con este trabajo: Blockchain, identidad digital e identidad soberana.
3. Análisis de las soluciones existentes para la gestión de la identidad en Blockchain.
4. Diseño y desarrollo de un método o aplicación que asegure la identidad o proporcione garantía sobre ella (sistema de identidad soberana).
5. Pruebas realizadas sobre el sistema desarrollado.
6. Análisis de la identidad soberana en el ámbito concreto de la medicina.
7. Puesta en marcha de un DLT para el intercambio de expedientes médicos.
8. Redactar la documentación final y la presentación del proyecto.

El proyecto comienza el día 17 de febrero del 2020 y finaliza el 2 de junio del mismo año.

Las tareas tienen las siguientes fechas de comienzo y final:

Name	Begin date	End date
Planficación del proyecto	17/2/2020	17/2/2020
Conceptos previos	18/2/2020	24/2/2020
Análisis de las soluciones existentes	25/2/2020	13/3/2020
Diseño y desarrollo SSI	16/3/2020	14/4/2020
Pruebas sobre el sistema	15/4/2020	16/4/2020
Análisis de la identidad soberana en la medicina	17/4/2020	21/4/2020
DLT para el intercambio de expedientes médicos	13/4/2020	28/4/2020
Documentación final y presentación	29/4/2020	2/6/2020

Figura 1 - Diagrama de Gantt - Tareas

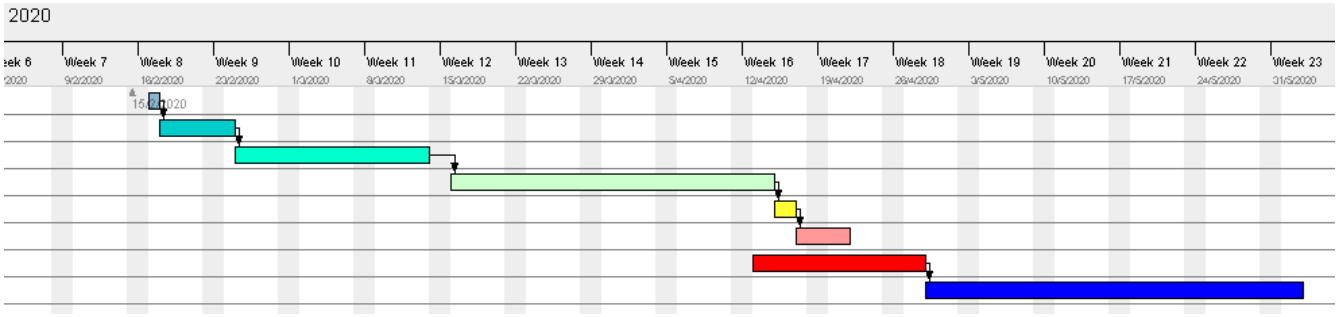

Figura 2 - Diagrama de Gantt - Línea temporal

3.3 Estimación de tiempos

En este apartado se puede visualizar una estimación de tiempos para las tareas que se han llevado a cabo en la realización del proyecto.

Tabla 1 - Estimación de tiempos

Iteración	Actividad	Horas
1	Realizar la planificación del proyecto	2
2	Introducción de conceptos básicos previos: Blockchain, identidad digital e identidad soberana	15
3	Análisis de las soluciones existentes para la gestión de la identidad en Blockchain	40
4	Diseño y desarrollo de un método o aplicación que asegure la identidad o proporcione garantía sobre ella (sistema de identidad soberana)	65
5	Pruebas realizadas sobre el sistema desarrollado	15
6	Análisis de la identidad soberana en el ámbito concreto de la medicina	15
7	Puesta en marcha de un DLT para el intercambio de expedientes médicos	50
8	Documentación y presentación del proyecto	23

TOTAL: 225 HORAS (9 CRÉDITOS)

Todas estas tareas hacen un total de 225 horas estimadas para el desarrollo del proyecto, trabajando una media de 3 horas al día en el proyecto, de lunes a viernes, serían 15 horas a la semana, que repartidas en 15 semanas (mediados de febrero - principios de junio) supondrían 225 horas, es decir, el tiempo requerido para el adecuado desarrollo del proyecto.

3.4 Estimación de costes

En la estimación de costes, aparte de considerar el factor humano, se debe considerar también el material y el software empleado en el desarrollo. En los costes materiales se han incluido el material de oficina y el ordenador que se ha utilizado durante el proyecto.

En la Tabla 2 se puede ver el desglose de precios:

Tabla 2 - Estimación de costes

Tipo de recurso	**Recurso**	**Coste de adquisición**
Rol/especialización	Salario	13 €/hora * 225 h = 2925 €
Material	Ordenador personal	1200 €
	Material de ofimática, soporte y papelería	150 €
	Total	**4275€**

3.5 Análisis de riesgos

A continuación, se enumeran una serie de riesgos que pueden alterar la planificación inicial del proyecto, así como las medidas de mitigación para cada posible riesgo.

R1: Falta de tiempo para el correcto desarrollo de todas las tareas definidas

Existen algunas tareas (especialmente las relativas a la parte práctica) que pueden causar problemas que conlleven cierto retraso, lo que podría implicar una falta de tiempo para el correcto desarrollo de todas las tareas definidas. También existen otros posibles motivos como el surgimiento de problemas personales o enfermedad que resten tiempo al proyecto.

Por lo tanto, si se considera necesario, se tendría que reducir el nivel de cada objetivo o incluso abandonar la puesta en marcha de alguno de ellos, poniendo el foco en el objetivo principal del proyecto y la viabilidad del mismo.

R2: Falta de tiempo para redactar la memoria del proyecto

Con la intención de cumplir con cada uno de las tareas del proyecto, puede que no se llegue a redactar la memoria a tiempo o que su elaboración cueste más de lo que se había planificado inicialmente.

Durante el desarrollo de proyecto es recomendable ir documentando todos los avances para reducir el tiempo necesario para la redacción de la memoria final.

R3: Problemas en el desarrollo

Pueden existir algunos problemas a la hora de realizar el desarrollo de un sistema de identidad soberana, ya que este ámbito de desarrollo es relativamente nuevo e inmaduro.

Si una solución de desarrollo de sistemas de identidad soberana, como por ejemplo Hyperledger Indy, resulta no ser adecuada para los desarrollos que se pretenden realizar en este proyecto, se podría emplear otra, como por ejemplo uPort. En caso de que los problemas persistan, podría requerirse la ayuda del director, co-director o de algún profesor colaborador del TFM.

3.6 Entregas parciales

El Trabajo Fin de Máster del Máster Universitario en Seguridad de las Tecnologías de la Información y de las Comunicaciones de la UOC se divide en una serie de entregables que permiten el seguimiento en forma de evaluación continua del proyecto.

Los entregables son:

PEC 1 (19/02/2020 - 03/03/2020): Plan de trabajo. Se trata de este mismo documento, donde se recoge la planificación detallada del proyecto (capítulos 1, 2 y 3).

PEC 2 (04/03/2020 - 31/03/2020): El entregable 2 contendría la parte teórica del TFM: análisis de los conceptos previos y análisis de las soluciones de identidad soberana existentes.

PEC 3 (01/04/2020 - 28/04/2020): El tercer entregable contendría la parte de diseño, desarrollo, puesta en marcha y testeo de una solución de identidad soberana, junto a un análisis exhaustivo de la identidad soberana en el ámbito de la medicina (intercambio de expedientes médicos).

Además, esta PEC contendrá la última y más importante tarea del proyecto: la puesta en marcha de un sistema de identidad soberana para el intercambio de expedientes médicos y la evaluación del sistema. Esta tarea principal se desarrollará a la vez que la tarea de diseño, desarrollo, puesta en marcha y testeo de una solución de identidad soberana, ya que dicha tarea principal no es más

que una adaptación para el intercambio de expedientes médicos del sistema diseñado y desarrollado inicialmente.

PEC 4 (29/04/2020 - 02/06/2020): El último entregable de la evaluación continua sería la memoria completa del proyecto, con todos los capítulos desarrollados en las PECs anteriores junto a otros como el resumen, conclusiones y trabajo futuro, glosario, referencias completas, anexos…

4. Conceptos previos

En este capítulo se van a introducir y explicar algunos conceptos básicos que hay que conocer con anterioridad para poder comprender de manera adecuada el proyecto. Dichos conceptos son: Blockchain, identidad digital e identidad soberana.

4.1 Blockchain

Blockchain ("cadena de bloques", en español) es un libro de contabilidad electrónico basado en una topología P2P (peer-to-peer) que se comparte abiertamente entre usuarios dispares para dar lugar a un registro inmutable de transacciones, cada una marcada en el tiempo y vinculada a la anterior. Cada vez que se agrega un conjunto de transacciones, esos datos se convierten en otro bloque en la cadena. [1]

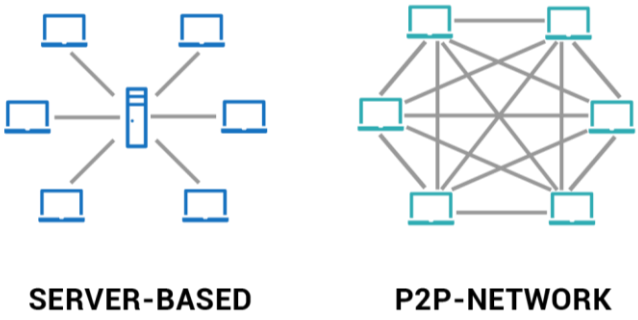

Figura 3 - Representación de la tipología usuario-servidor y P2P [16]

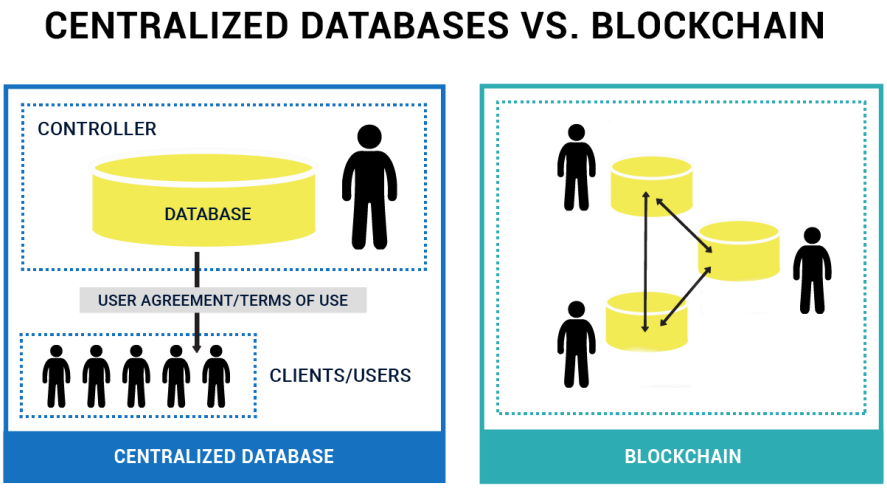

Figura 4 - Diferencia entre una BD centralizada y Blockchain [17]

Cada bloque contiene un hash criptográfico del bloque anterior, un hash propio, una marca de tiempo y los datos de la transacción. Debido a esto, la información contenida en un bloque solo puede ser repudiada o editada modificando todos los bloques posteriores, haciendo muy difícil la manipulación maliciosa de la información. [2]

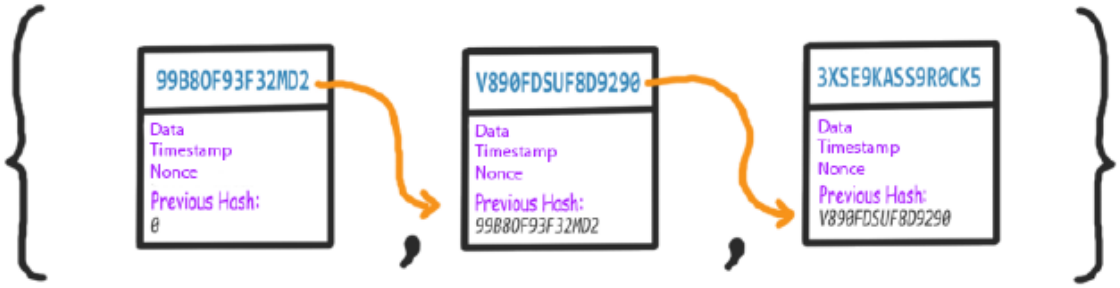

Figura 5 - Representación de una cadena de bloques básica

Además, en los sistemas Blockchain públicos (ver apartado 4.1.1), como las criptodivisas, se suelen emplear protocolos diseñados para evitar comportamientos maliciosos. El protocolo más conocido es la Prueba de Trabajo (Proof of Work). Cada bloque de la cadena de bloques contiene un dato, típicamente un número entero, llamado "nonce" (number used once). En un sistema Blockchain público que emplea un protocolo de validación competitivo como la PoW, existen una serie de actores llamados "mineros" que emplean una gran cantidad de recursos para encontrar un "nonce" que dé solución a un problema matemático muy complejo de hallar, pero fácilmente comprobable por los demás. Específicamente, se requiere que el hash SHA256 del bloque tenga un cierto número de ceros iniciales. Los hashes son funciones unidireccionales, por lo que existe ninguna forma de conocer el "nonce" correcto o de diseñar de cualquier modo el bloque correcto. La única forma de encontrar un buen nonce es realizar muchas iteraciones hasta dar con el correcto. [3] Debido a que este cálculo iterativo requiere tiempo y recursos, el hallazgo del bloque con el valor "nonce" correcto constituye una prueba de trabajo, por lo que es el minero que encuentre primero dicha solución el que añadirá el bloque a la cadena, esperará la validación de dicho bloque por parte de los demás mineros y además recibirá una recompensa por su "trabajo". En conclusión, la misión de la minería es certificar que nadie usa las monedas dos veces y que nadie pueda introducir en el mercado monedas falsas (si se trata de una criptodivisa). [4]

4.1.1 Tipos de Blockchain

Existen 3 tipos principales de cadenas de bloques: públicas, privadas e híbridas.

Una Blockchain pública es una cadena de bloques accesible a cualquier usuario. Los ejemplos más conocidos de blockhains públicas son las criptomonedas como Bitcoin o Ethereum, donde cualquier actor puede unirse a la red, incluyendo actores malintencionados.

En el caso de las cadenas de bloques privadas, el número de participantes está limitado. Las redes permisionadas como Hyperledger, plataforma que se utiliza en este proyecto, reducen el riesgo dejando unirse a la red solo a participantes conocidos que deben ser aprobados por el resto de participantes. Si los participantes de una Blockchain privada descubren que uno de los participantes está actuando malintencionadamente pueden denegarle el acceso por medio de un consenso, por lo que en el caso de este tipo de Blockchain, no se requieren protocolos de validación muy complejos. [5]

Una Blockchain híbrida es una cadena intermedia que trata de aprovechar las ventajas de las públicas y de las privadas. Idealmente, una Blockchain híbrida significará acceso controlado y libertad al mismo tiempo. Las cadenas híbridas se distinguen por el hecho de que no están abiertas para todo el mundo, pero aun así ofrecen funcionalidades típicas de Blockchain como la integridad, la transparencia y la seguridad. Las Blockchain híbridas son totalmente personalizable. Los miembros de una cadena híbrida pueden decidir quiénes pueden participar o qué transacciones se hacen públicas. [7]

4.2 Identidad digital

La identidad digital, también conocida como “identidad 2.0”, es todo lo que le identifica a un sujeto en el entorno online. En otras palabras, es el conjunto de informaciones publicadas en Internet sobre un sujeto y que componen la imagen que los demás tienen de él: datos personales, imágenes, noticias, comentarios, gustos, amistades, aficiones, etc. Todos estos datos determinan la reputación digital de cada persona.

Esta identidad puede no corresponderse de manera exacta con la realidad. Sin embargo, esa identidad digital tiene sus consecuencias en el mundo real y viceversa. En definitiva, la identidad digital es la traslación de la identidad física al mundo online. [8]

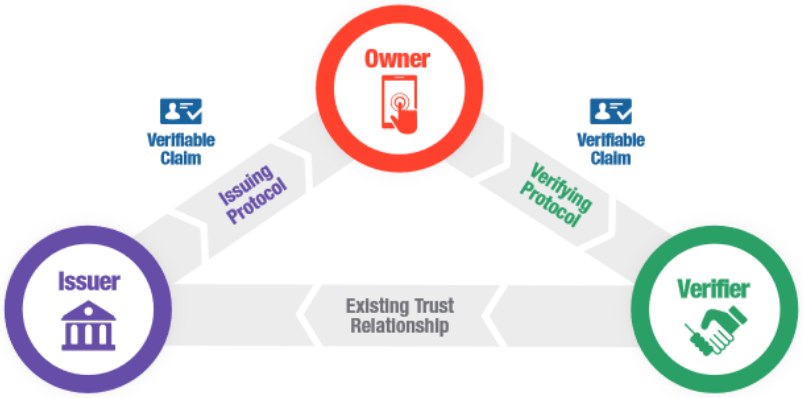

Figura 6 - Representación gráfica del concepto de identidad [30]

La identidad digital se compone de datos de diferente procedencia como, por ejemplo [9]:

- Perfiles personales. Redes sociales generales (Facebook, Instagram, Twitter…) y profesionales (LinkedIn, Viadeo, Xing…) y portales de búsqueda de empleo como Infojobs.
- Comentarios. En foros, blogs, portales de información, redes sociales, YouTube….
- Contenidos digitales. Fotos en redes sociales, videos, presentaciones en Slideshare o documentos publicados en webs, una web personal, un blog…
- Contactos. Amigos, contactos profesionales, seguidores y personas u organizaciones seguidas…
- Las direcciones de correo electrónico.
- La mensajería instantánea. Messenger, Telegram, Whatsapp…

4.3 Identidad soberana

Uno de los conceptos que más popularidad está ganando dentro del mundo de Blockchain es la identidad soberana.

La identidad soberana es un tipo de identidad digital en el que el usuario tiene pleno control sobre sus datos. Un sistema de identidad soberana permite al usuario manejar quienes pueden acceder a ellos y en qué términos. Este concepto revolucionario ha captado la atención de varias organizaciones, individuos e incluso empresas. [10]

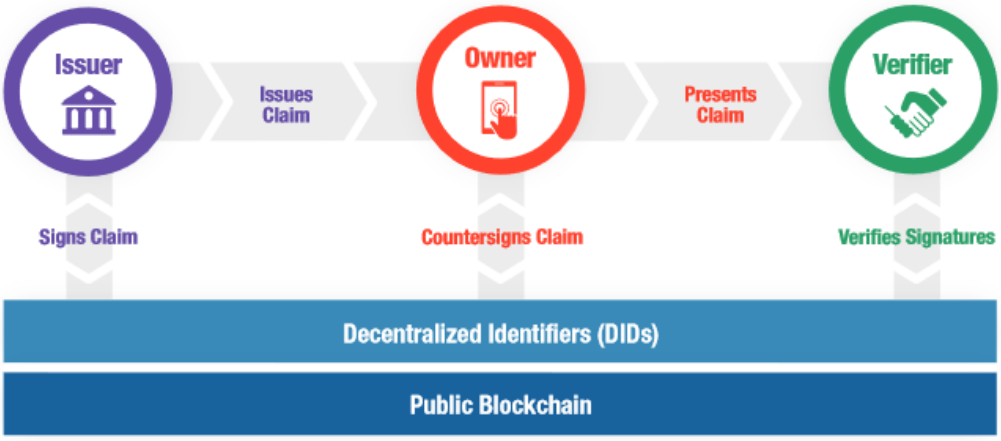

Figura 7 - Representación gráfica de un sistema de identidad soberana (SSI) [30]

Las soluciones de identidad actuales exigen poseer una gran variedad de tarjetas de plástico que usamos para identificarnos o hacer pagos. En cuanto al ámbito de la red, se necesita tener un nombre de usuario y contraseña para muchos sitios web. Esto implica que las personas tienen poco o ningún control sobre sus datos. Además, los servidores que guardan o recopilar datos sensibles se convierten objetivos de ataques maliciosos. Todavía existe un problema de identidad digital, aunque ya existen múltiples propuestas para solucionar dicho problema. Una de ellos es la identidad soberana. [11]

Se considera que la mejor base para la correcta implementación de la identidad soberana es Blockchain, ya que se trata de un concepto fuertemente relacionado con la criptografía, la descentralización y la seguridad que proporciona la tecnología Blockchain. [10] De hecho, al aprovechar una estructura Blockchain, se puede observar que algunas de las propiedades de la identidad soberana se cumplen intrínsecamente [11]:

- **Existencia.** Los usuarios deben poseer una existencia independiente.
- **Control.** Los usuarios deben controlar sus identidades.
- **Acceso.** Los usuarios deben tener acceso a sus propios datos.

- **Transparencia.** Los sistemas y algoritmos deben ser transparentes.
- **Persistencia.** Las identidades deben ser duraderas.
- **Portabilidad.** Información y servicios sobre la identidad debe ser transportable.
- **Interoperabilidad.** Las identidades deben ser tan ampliamente utilizables en la medida de lo posible.
- **Consentimiento.** Los usuarios deben estar de acuerdo con el uso de su identidad.
- **Minimización.** La divulgación de los reclamos debe ser minimizada.
- **Protección.** Los derechos de los usuarios deben ser protegidos.

4.3.1 ¿Cómo funciona?

En un sistema de identidad soberana, el usuario dueño de la identidad posee en todo momento un control total y soberano sobre su identidad. Los datos de identidad se almacenan en un formato protegido por criptografía asimétrica (clave pública y clave privada). Así, el usuario puede compartir datos con terceros de forma segura y sin exponerse a fugas de datos no deseadas.

Adicionalmente, el usuario tiene control de cada transacción de su información. En otras palabras, cada intercambio de datos se produce en los términos que el usuario decide. Es el usuario es quien decide qué información compartir, cuánta y con quién. Este nivel de control es el principal factor diferenciador entre los sistemas de identidad digital centralizada o federadas y los sistemas de identidad soberana.

Finalmente, la compartición de la información se da sobre un sistema totalmente descentralizado (Blockchain). En este sistema, cada participante está capacitado por medio de un consenso de establecer si los datos de identidad otorgados son ciertos o falsos. Al igual que en el caso de las criptodivisas no existe un banco central que las controle o regule, en el caso de la identidad soberana no existe una autoridad central, que pueda dictar reglas o censurar acciones. Con un sistema de consenso se busca garantizar que los datos proporcionados no estén manipulados de una u otra forma. [10]

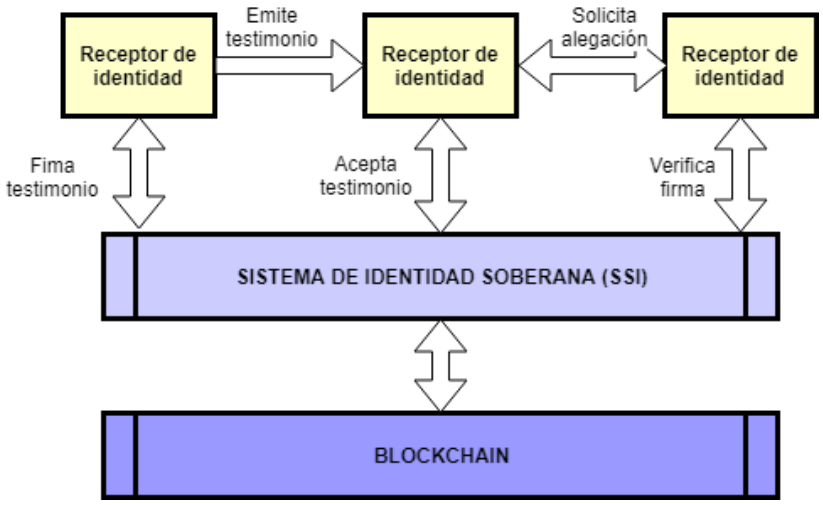

Figura 8 - Funcionamiento de la identidad soberana [12]

4.3.2 Razones para desarrollar y utilizar sistemas de identidad soberana

Las principales razones para diseñar y utilizar sistemas de identidad soberana en contraposición a un sistema centralizado son las siguientes:

Seguridad

La mayoría de los sistemas de identidad digital que existen actualmente son de tipo centralizado. Estos sistemas se basan en grandes bases de datos centralizadas que contienen millones de registros de identidad. Debido a su tamaño y a la información almacenada, estos almacenes de datos son objetivos muy valiosos para los piratas informáticos. Por lo tanto, mientras más identidades contenga una base de datos, más riesgo existe de que sea atacada. [10]

Control de los datos

Un sistema de identidad soberana hace que los usuarios puedan poseer y controlar plenamente sus datos de identidad. Además, desaparece la necesidad de confiar en una autoridad única y central. Se aseguraría la seguridad y la privacidad de los datos. [13]

Para las organizaciones: Facilidad, reducción de costes y aumento de la confianza

El uso de la identidad soberana podría ayudar a reducir los costes relacionados con la gestión de identidades por parte de los gobiernos. Además, un sistema SSI ofrece una total transparencia en la gestión de la identidad, lo que puede ayudar a aumentar la confianza de los ciudadanos en las

instituciones de sus Estados. Además, los procesos y los servicios gubernamentales transfronterizos pueden realizarse más fácilmente.

Las empresas privadas también pueden beneficiarse de un sistema de SSI a la hora de prestar servicios y al requerirse un proveedor de identidad calificado. Esto puede suponer un ahorro de tiempo y costes a las empresas. [13]

4.3.3 Desafíos y limitaciones

Cuando se aplica un modelo SSI, surgen varios retos como el nivel de permiso de acceso a la cadena de bloques, los cálculos necesarios para la prueba de trabajo, la falta de comprensión técnica y los problemas de almacenamiento de los datos. [13]

El nivel de permiso de acceso de la cadena de bloques depende de la tecnología elegida. Puede variar entre el nivel de permiso de acceso público y el privado. El uso de una cadena de bloques pública requiere la implementación de un algoritmo de consenso como la prueba de trabajo evitar comportamientos maliciosos. La prueba de trabajo requiere una gran cantidad de potencia de cálculo. Por el contrario, el uso de una cadena de bloques privada requiere confianza en las partes que son responsables de escribir y leer los datos desde o hacia el libro de contabilidad.

Otro obstáculo cuando se utiliza el modelo SSI puede ser la falta de comprensión técnica de esta novedosa tecnología. Esta falta de conocimiento puede dar lugar a problemas de integración cuando se intenta integrar un modelo SSI en una infraestructura existente.

La utilización de SSI también plantea problemas de almacenamiento de datos, especialmente cuando se trata de datos sensibles. Algunas implementaciones de cadenas de bloques utilizan almacenamiento externo adicional. Cuando los datos confidenciales se almacenan cifrados en la cadena de bloques, también surgen problemas de distribución de claves. La capacidad de almacenamiento limitada de las blockchains describe otro problema de almacenamiento.

4.3.4 Casos de uso de la identidad soberana

Los casos de uso donde se podría emplear un SSI son muy variados. A continuación, se exponen algunos casos de uso que podrían ser especialmente útiles e interesantes.

La identidad soberana es ideal para la acreditación de certificados oficiales, como podrían ser las titulaciones académicas. Así pues, se podría vincular a la identidad de un usuario con un certificado educativo criptográficamente seguro. Este concepto ya está siendo usado por el Instituto de Tecnología de Massachussets (MIT). [10] Este sistema recibe el nombre de "Proyecto de Certificados Digitales" y hace uso de la Blockchain de Bitcoin. En dicha cadena de bloques se almacena la información del estudiante y su certificado como una prueba irrefutable de que este ha recibido dicho certificado. En adición a este sistema, existen otros sistemas que permiten la revisión y verificación por medio de un ID de los certificados emitidos por la institución educativa.

Otros casos de uso muy interesantes para la identidad soberana sería el experimento de la ciudad suiza de Zug. A finales de 2017, dicha cuidad lanzó oficialmente un sistema de identidad soberana basado en Ethereum para todos sus residentes. [14] Mediante este sistema, los ciudadanos de Zug han podido utilizar servicios electrónicos como la votación en línea, aportar prueba de residencia, y más recientemente, un servicio para compartir bicicletas. Este último caso representa una apertura a un amplio abanico de oportunidades para que los ciudadanos "verificados" mediante el sistema SSI puedan acceder a servicios del sector privado. Los usuarios no solo pueden verificar que las personas que acceden a sus bicicletas son personas confiables, es decir, residentes de Zug, sino que tienen la posibilidad de tener un control total sobre sus datos.

Finalmente, otro sector clave donde la identidad soberana podría encajar es el sector de la salud. [15] Dicho sector es un campo donde la inalterabilidad y el control de la información son fundamentales. Actualmente, la información crítica de los pacientes suele a estar dispersa en varios sistemas y, a veces, no es accesible en momentos críticos. Es habitual que centros privados y públicos guarden y no compartan su información. De esta forma, cada vez que un usuario cambia de centro pierde todo su historial anterior. Mediante el uso de la identidad soberana en el ámbito de la salud, el paciente recibiría mediante su identidad digital la información médica que los centros producen y sería decisión suya si compartirla, con quién compartirla y durante cuánto tiempo compartirla. Esto no solo le facilitaría al paciente la gestión y el control sobre sus datos médicos, sino que también podría mejorar la investigación científica, al ser más sencillo colaborar con los investigadores en áreas de interés.

4.3.5 GDPR y la identidad soberana

El 25 de mayo del año 2018 comenzó la aplicación del Reglamento General de Protección de Datos (GDPR), que busca la protección de las personas físicas en lo que respecta al tratamiento de sus datos personales y a la libre circulación de estos datos. Se trata de una normativa a nivel de la Unión Europea, por lo que cualquier organización de la unión, o aquellas organizaciones que tengan negocios en la Unión Europea, que manejen información personal de cualquier tipo, deberán respetarla. Las multas por el incumplimiento del Reglamento pueden alcanzar los 20 millones de euros. [18]

Un sistema de SSI puede ayudar al cumplimiento del GDPR cuando se trata de IDs digitales. A continuación, se detalla cómo el sistema SSI puede cumplir con algunos artículos del GDPR. [13]

Consentimiento

El GDPR especifica que un usuario debe dar su consentimiento explícito para que una organización pueda recoger y procesar sus datos. Un sistema SSI puede implementar una interfaz gráfica optimizada para el usuario. Esta interfaz proporcionaría una visualización de los datos que se estarían compartiendo junto con mecanismos de revocación.

Seudonimización

El GDPR describe la seudonimización como un proceso en el que los datos personales de los ciudadanos se transforman de tal manera que los datos resultantes no podrían ser vinculados a una persona específica.

En un sistema SSI, sólo se almacenan identificadores en la cadena de bloques. Estos identificadores se generan mediante criptografía y no pueden ser vinculados a una persona específica.

Derecho al olvido

El derecho al olvido del GDPR describe el derecho de un ciudadano de la UE a solicitar la supresión de sus datos personales. En un sistema SSI, el usuario es propietario de todos sus datos

de identidad; por lo tanto, el usuario puede simplemente borrar toda la identidad y sus datos relacionados. El borrado se realiza no almacenando ningún dato privado en un lugar de acceso público como el propio libro de contabilidad descentralizado. En su lugar, en el libro sólo se almacenan los identificadores, vinculados a los datos de identidad.

Portabilidad de datos

En el GDPR, el derecho de portabilidad de datos describe el derecho que tiene una persona dentro de la UE de transferir sus datos personales de un lugar a otro. El sistema SSI apoya este derecho proporcionando una capa de identidad abierta para Internet, que ofrece la posibilidad de acceder y utilizarla en todo el mundo. Esta posibilidad existe debido a que el sistema SSI combina diferentes tecnologías que permiten la portabilidad de los datos, como el libro mayor distribuido (DLT), con formatos de intercambio de datos estandarizados.

Protección de datos por diseño y por defecto

La protección de datos por defecto significa que los mecanismos de protección de datos ya forman parte del diseño del sistema, como es el caso del sistema SSI. Esto implica que, por defecto, unas medidas técnicas y organizativas adecuadas deben garantizar la protección de los datos personales relacionados con un ciudadano de la UE.

El sistema SSI aplica técnicas de vanguardia tanto para preservar la privacidad del usuario como para proteger los datos que están siendo tratados. Una de estas técnicas de vanguardia es la Prueba de Conocimiento Cero (ZKP), un protocolo criptográfico que establece un método para que una de las partes pruebe a otra que una declaración es cierta, sin revelar nada más que la veracidad de la declaración. [19] Por ejemplo, mediante una ZKP se puede demostrar la posesión de un permiso de conducir sin revelar dicho documento completo.

5. Estudio de las soluciones existentes de identidad soberana

5.1 Alastria

5.1.1 Introducción

Alastria es un ecosistema español basado en Blockchain y enfocado a las empresas que optan por digitalizar sus activos. En la red Alastria, los participantes pueden establecer representaciones digitales de los activos con los que trabajan. Dichos activos se representan como "tokens". Las empresas pueden utilizar dichos tokens para poner en marcha nuevos productos y servicios, o para optimizar sus procesos actuales. Alastria emplea la plataforma de Ethereum para funcionar. [20]

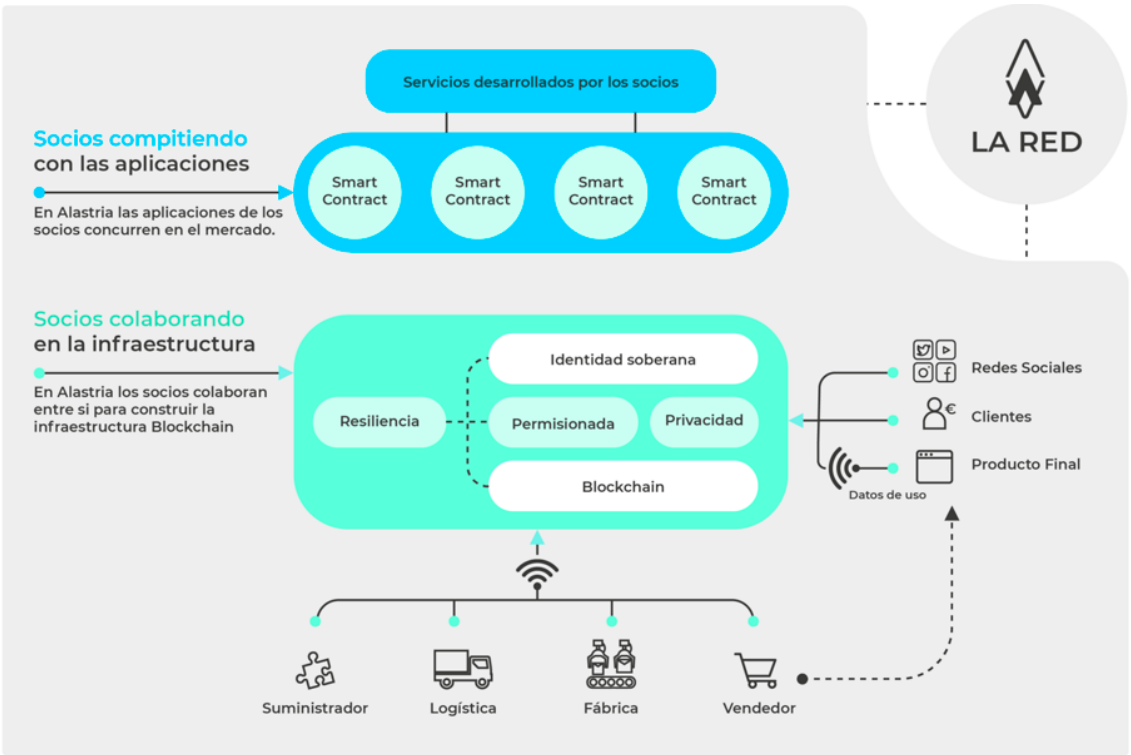

Figura 9 - La Red Alastria [23]

Alastria es una combinación de cadenas de bloques semipúblicas y autorizadas. El consorcio funciona con un sistema basado en membresía. Existen tres niveles diferentes de membresía,

dependiendo del tipo de negocio que desea participar y su tamaño. Las pymes abonan una tarifa de 500€ al año, mientras que las empresas medianas pagan 5.000€ al año y las más grandes, 10.000€. No existe ninguna cuota para las organizaciones sin ánimo de lucro, pero es necesario que estas firmen un convenio para poder convertirse en miembros. Al ser una organización sin ánimo de lucro, Alastria usa todos sus ingresos para financiar el futuro desarrollo de Blockchain. [21]

Las bases de Alastria

Según la página oficial de Alastria [22], estas son las bases del proyecto:

- **Identidad Digital:** La Red Alastria también proporciona una herramienta de gestión de la identidad digital, asunto que se trata en detalle en el apartado 5.1.2 de este documento.
- **Virtualizar los activos:** La Red Alastria permite representar los activos empresariales en la Blockchain mediante tokens para agilizar las transacciones y transferencias.
- **Smart Supply Chain:** Simplificar procesos que a día de hoy son complejos.
- **Economía Digital:** Gestionar y facilitar las relaciones entre empresas.
- **Hub de Innovación:** Crear un ecosistema para atraer el talento. Creación de espacios en distintas ciudades españolas para formar futuros desarrolladores en Blockchain.

Principales actividades de Alastria

Según la página oficial de Alastria [23] [4], estas son las principales actividades y objetivos del consorcio:

- Establecer los estándares técnicos de la infraestructura denominada “Red Alastria”, y promover el acuerdo entre los asociados para su desarrollo, explotación y uso.
- Promover acuerdos entre sus asociados para establecer y organizar servicios comunes para los propios asociados y quienes utilicen la Red Alastria.
- Facilitar que los asociados puedan proponer y promover el desarrollo y explotación de aplicaciones soportadas en Red Alastria.
- Fomentar el conocimiento y el uso por la sociedad española de las tecnologías tipo DLT o Blockchain, promoviendo su uso entre las Administraciones, empresas y demás agentes sociales.

- Representar colectivamente a sus miembros, siempre en el ámbito de los fines y actividades de esta Asociación ante terceras personas y organizaciones, planteando las actividades necesarias para el cumplimiento de estos fines.

Figura 10 - Principales objetivos de Alastria [29]

Los nodos de la Red Alastria

En la Red Alastria existen dos tipos de nodos: los nodos validadores y los nodos observadores.

Los Nodos Validadores ejecutan el protocolo de consenso para validar las nuevas transacciones de Alastria. Cada "escritura" al libro mayor de Alastria se lleva a cabo por un nodo de validación. [24] El minado de Alastria funciona de forma rotatoria, un nodo crea el bloque, lo propone a los demás y lo introduce de forma definitiva a la cadena si las dos terceras partes de los demás nodos dan el visto bueno. Este proceso lleva poco más de un minuto, y se está intentado hacer que llegue a unos cuantos milisegundos.

Los Nodos Observadores se necesitarán a medida que la red se desarrolle, para distribuir la carga de lectura sobre la cadena de bloques, ayudando así a garantizar la confidencialidad y la privacidad de la información. [25]

Figura 11 - Representación de los Nodos Alastria [4]

5.1.2 Alastria ID

ID Alastria es un modelo de identidad digital basado en el concepto de Self Sovereign Identity (SSI) diseñado por el Consorcio de Alastria para ser empleado en servicios digitales. [26]

La identidad digital es el foco principal de Alastria en la actualidad. Este sistema denominado "Alastria ID" permite a los ciudadanos tener el control total sobre sus datos personales de manera transparente siguiendo las pautas establecidas por la Unión Europea. Con la puesta en marcha del "ID Alastria" los usuarios serán los únicos titulares y custodios de sus datos de identidad. Este sistema busca la unificación de todas las identidades digitales que puede llegar a poseer un usuario, en una sola, más privada y segura. En definitiva, Alastria ID busca dotar de validez legal a las relaciones entre los usuarios de la red manteniendo al mismo tiempo la privacidad y seguridad de los datos. [25]

Figura 12 - Representación visual del funcionamiento del ID Alastria [4]

El modelo de Alastria ID

El modelo de Alastria ID se basa en tres roles principales [28]:

1. **Issuers.** Se trata de las entidades que certifican determinados datos acerca de los usuarios. Por ejemplo: La UOC puede ser un "issuer" que certifique que la obtención del Título de Máster Universitario en Seguridad de las TIC en cierta fecha.
2. **Service Providers.** Se trata de los proveedores con los que el usuario desea compartir ciertos datos para recibir ciertos servicios. Por ejemplo: Una empresa de alquiler de coches necesita conocer la fecha en la que el usuario ha obtenido su carnet de conducir.
3. **User**s. Se trata de los usuarios que poseen sus datos, deciden con quién compartirlos y cuáles de ellos compartir.

Este modelo posee enormes ventajas, ya que son los usuarios quienes controlan su propia información y, además, una vez dichos usuarios poseen esta información certificada, pueden volver a utilizarla de forma constante sin necesidad de rellenar múltiples formularios o papeles cada vez que deseen realizar alguna operación.

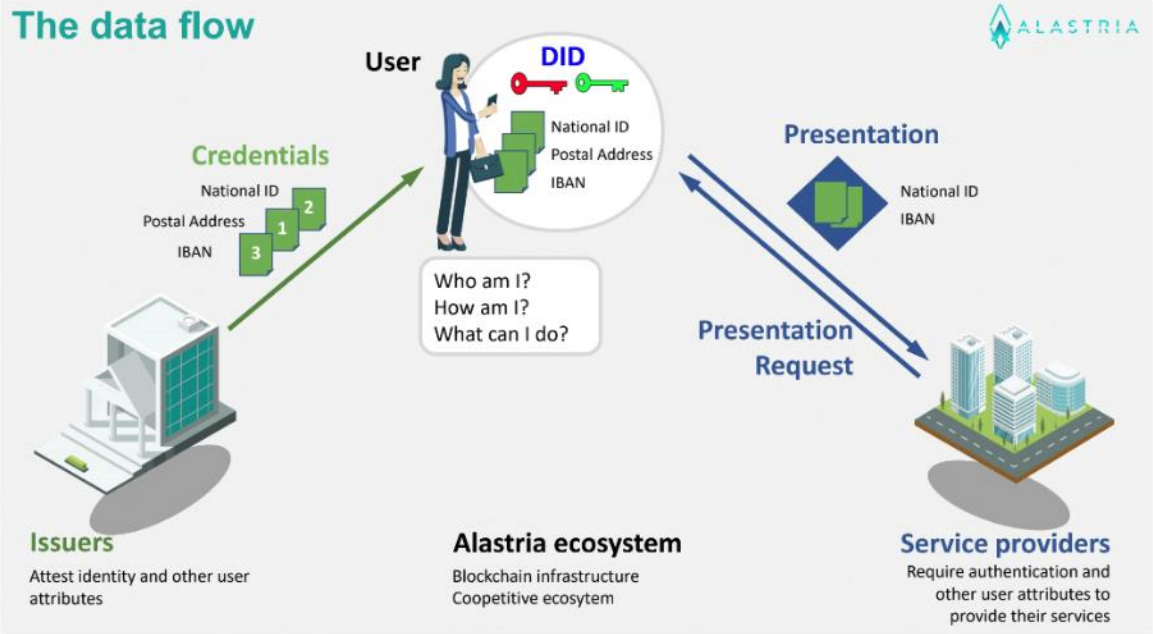

Figura 13 - Representación gráfica del modelo de Alastria ID [28]

La infraestructura de Alastria ID

Según la presentación oficial del proyecto [27], Alastria ID cuenta con la siguiente estructura y actividad:

- **Aplicación.** Es la aplicación a través de la cual el usuario interactúa con Alastria ID, almacena y custodia su clave privada y controla el repositorio de datos personales.
- **Repositorio.** Se trata de un repositorio ubicado fuera de la cadena de bloques (off-chain), de elección y bajo control del usuario donde administrarían y almacenarían los elementos de información del modelo: claves, alegaciones, testimonios y documentos asociados, que podrían ser igualmente verificados.
- **Blockchain Alastria.** Es donde se almacenan la clave pública del usuario y los hashes de los testimonios y alegaciones, es decir, las evidencias de existencia que impiden la alteración, así como las revocaciones de los mismos. Tal y como se ha especificado anteriormente en la definición de identidad soberana (SSI), en la cadena de bloques nunca se almacenan directamente los propios datos personales.

A continuación, se muestra una representación gráfica de la infraestructura descrita anteriormente:

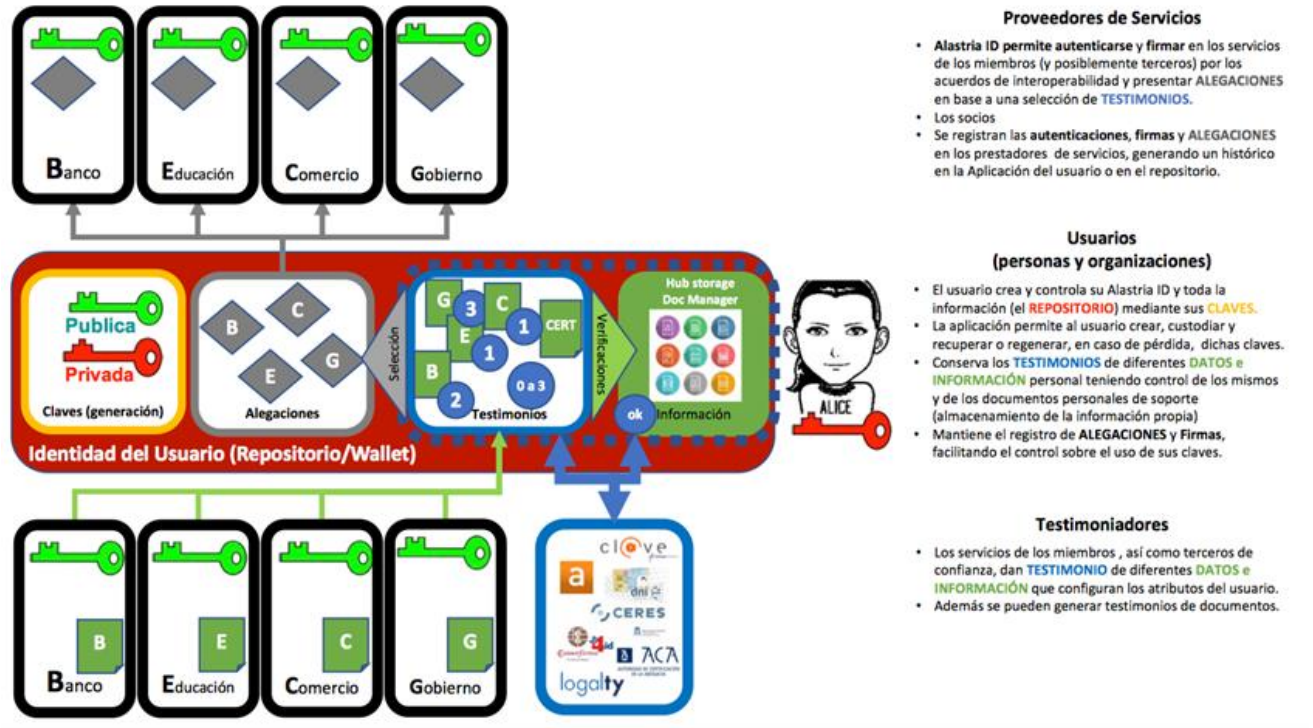

Figura 14 - Infraestructura de Alastria ID [27]

En el Anexo II “Información adicional” se pueden consultar, paso a paso, el proceso que se lleva a cabo al realizar una acción en Alastria ID (creación de la identidad, transacciones, firma, etc.).

5.2 Sovrin

5.2.1 Introducción

Sovrin es un protocolo y un token de identidad soberana y de confianza descentralizada de tipo *open-source*. Es una cadena de bloques autorizada que pertenece y es dirigida por la Fundación Sovrin. Dicha fundación tiene la intención de implementar sistemas de identidad soberana (SSI), alcanzando así un Internet en el que las personas controlen y custodien la información digital que las identifica, pudiendo elegir cuándo y cómo compartirla sin tener que depender de un tercero, como las empresas tecnológicas o los gobiernos. Sovrin contempla una identidad digital permanente, portable, privada y completamente segura.

Cabe destacar que los proyectos de Sovrin y Hyperledger Indy (apartado 5.3) están estrechamente relacionados debido a que Indy está basado en el código liberado de Sovrin, por lo que el funcionamiento de ambas plataformas será idéntico.

Los usuarios de la plataforma pueden poseer una identidad soberana que pueden usar para gestionar una variedad de IDs como: billetes de avión, licencias académicas, permiso de conducir, etc. Una vez almacenada en la cadena de bloques, la identidad de cada usuario está protegida ante posibles accesos no autorizados o intentos de manipulación. [31]

La arquitectura de Sovrin establece la división de los nodos en 2 partes: un anillo de nodos validadores para las transacciones de escritura, y un anillo mucho más amplio de nodos observadores que ejecutan copias de sólo lectura de la cadena de bloques para procesar solicitudes de lectura. El primero es responsable de almacenar la información de la transacción, y el segundo es responsable de alcanzar el consenso, por lo que se reduce la necesidad de contar con muchos nodos y la velocidad de verificación aumenta. Actualmente, la Red Sovrin cuenta con más de 30 nodos totalmente funcionales. [33]

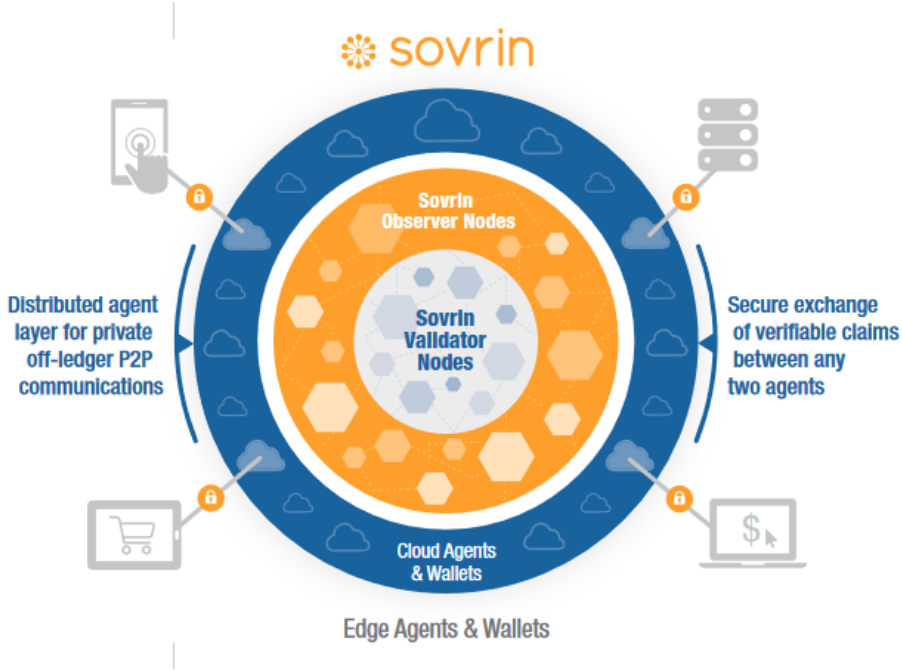

Figura 15 - Nodos de Sovrin [30]

El libro mayor de Sovrin, al ser de tipo autorizado, no requiere ningún algoritmo de consenso costoso como la prueba de trabajo (PoW), lo que reduce significativamente el coste de energía en el funcionamiento de un nodo y mejora drásticamente el rendimiento de las transacciones. En segundo lugar, la confianza en Sovrin depende tanto de la gente como del código fuente. La confianza empieza en la raíz común de la confianza establecida por el libro mayor distribuido, y a medida que nuevas organizaciones y usuarios se unen a la red, pueden convertirse en anclajes de confianza (es decir, se les permite añadir más usuarios y organizaciones); es decir, se espera que se desarrolle una "red de confianza" que logre la descentralización de la red.

Sovrin cuenta con una aplicación móvil que sirve para que los usuarios puedan gestionar sus claves criptográficas, que se almacenarían dentro del propio dispositivo móvil del usuario.

Sovrin ofrece un mecanismo para la recuperación de claves que se basa en la selección de algunos agentes de confianza. Cuando el usuario lo solicita, un quórum específico de agentes de confianza debe firmar una nueva transacción de registro de identidad que queda pendiente de la verificación por parte de unos administradores. [34] Todos estos roles se explicarán más en detalle a continuación.

Roles oficiales en la infraestructura de Sovrin

La infraestructura de Sovrin es mantenida por tres tipos de actores. Nótese que una sola organización puede desempeñar las tres funciones. [35]

1. Los **administradores (stewards)** son organizaciones que operan los nodos del Sovrin Ledger. Los administradores iniciales son nombrados por el Consejo de Administración de la Fundación Sovrin, lo que hace que Sovrin sea un libro de contabilidad autorizado. Dichos administradores deben firmar el “Convenio de Sovrin Steward” de la Fundación Sovrin.
2. Las **agencias** son organizaciones que hospedan “agentes” en nombre de los propietarios de las identidades. Las agencias que desean ser reconocidas oficialmente por la Fundación Sovrin deben firmar el “Acuerdo de Agencia Sovrin” de la Fundación Sovrin.
3. Los **desarrolladores** son organizaciones o individuos que desarrollan aplicaciones que usan la Red Sovrin. Los desarrolladores que deseen ser reconocidos oficialmente por la

Fundación Sovrin deben firmar el “Contrato de Desarrollador de Sovrin” de la Fundación Sovrin.

5.2.2 Aspectos técnicos de Sovrin

En este apartado se van a explicar detalladamente los detalles técnicos avanzados que contiene el sistema de Sovrin. La mayor parte de las explicaciones presentes en este apartado están basadas en un White Paper oficial de la Sovrin Foundation junto con Evernym: “*White Paper: Sovrin: What Goes On The Ledger?*” [33]

Si se desea consultar el funcionamiento de Sovrin resumido, consulte el apartado 5.2.3.

Identificadores descentralizados (DID)

Sovrin utiliza identificadores digitales descentralizados (DID) para llevar a cabo la autenticación de la identidad, mientras que la privacidad se garantiza por su diseño de red. A diferencia de las firmas tradicionales que utilizan un mismo identificador, los DID utilizan identificadores en forma de seudónimos por defecto para reducir la relevancia de cada identificador. De esta manera, se evita el almacenamiento de claves privadas en la cadena de bloques. [32] Dado que el documento DID contiene la clave de verificación actual (también conocida como clave pública) utilizada por el titular de la identidad, y dado que el Libro Mayor de Sovrin proporciona una verificación criptográfica del documento DID actual, los usuarios pueden confiar en que Sovrin devolverá la clave de verificación actual para cualquier DID. Al hacer esto sin necesidad de depender de ninguna autoridad centralizada, se resuelve uno de los mayores problemas de la PKI (infraestructura de clave pública) existente: saber qué clave es una autoridad en el momento para un titular de identidad específico.

Un documento DID también contiene un puntero a un endpoint de servicio. El endpoint es la dirección de red que el titular de la identidad utiliza para la comunicación posterior, como una URL o una dirección IP, y permite que dos titulares de identidad se comuniquen directamente entre sí en una interacción privada y segura entre pares, DID a DID, sin intermediarios. Un agente es un programa o proceso de software que actúa en nombre de un titular de identidad para facilitar las interacciones con otros agentes o con el Libro Mayor de Sovrin. Con un DID de

Sovrin, un agente tiene una dirección persistente para enviar, recibir, almacenar o encaminar mensajes cifrados a su titular de identidad (muy similar al correo electrónico o la mensajería instantánea, solo que en este caso todo el proceso está cifrado y es descentralizado).

La utilización de un único identificador para todas las transacciones afecta gravemente a la privacidad del titular de la identidad, ya que se podría establecer fácilmente una correlación entre todas sus transacciones. En las aplicaciones de libros distribuidos en las que esas transacciones se escriben permanentemente en el libro de contabilidad, este enfoque da lugar a un registro público inmutable de todas las interacciones de un titular de identidad. En el caso de Sovrin, cada titular de una identidad establece un DID único para cada una de sus relaciones. Estos DID únicos reciben el nombre de “DID seudónimos por pares”. El titular de la identidad es la única persona capaz de mapear y correlacionar sus DID seudónimos por pares (cada uno de los cuales también tiene una clave pública y un endpoint de agente seudónimos por pares). [33] En la práctica esto significaría que la relación que un usuario establece con una institución “X” está completamente separada de la relación que establece con una institución “Y”, por lo que ni “X” ni “Y” podrían utilizar los DID para correlacionar las actividades del titular de la identidad con el otro.

DIDs públicos y privados

Los usuarios que utilizan la red de Sovrin también tienen la posibilidad de emplear DID públicos o privados. [33]

Un DID que se añade directamente al libro público de Sovrin se denomina DID público, mientras que un DID anónimo compartido y almacenado de forma privada "fuera del libro" entre los agentes de los titulares de dos identidades se denomina DID privado.

Los DID públicos son necesarios ante todo para los emisores de credenciales. Se almacenan en el libro mayor público de Sovrin para que un verificador que recibe la prueba de cualquier credencial pueda buscar la clave pública del emisor (además de la definición de la credencial y el registro de revocación) para llevar a cabo la validación.

Por el contrario, los DID privados se utilizan por defecto en las relaciones privadas entre dos titulares de identidad, donde nadie más necesita conocer dichas relaciones, y tampoco los DID

que se están utilizando. Por lo tanto, no es necesario que los DIDs privados sean almacenados en el libro de contabilidad público de Sovrin. Si una parte cambia su clave pública o el endpoint del agente, simplemente le tiene que pasar a la otra parte las nuevas claves y/o el nuevo endpoint.

El hecho de poder mantener los DIDs privados y las comunicaciones entre DIDs fuera del libro público ayuda a reducir drásticamente la carga y sirve para aprovechar al máximo las infraestructuras de computación en la nube de hoy en día, pudiendo proteger la privacidad y la integridad de cada titular de la identidad al mismo tiempo.

Esquemas y definición de credenciales

Para apoyar el intercambio interoperativo de credenciales verificables, el libro mayor de Sovrin almacena dos objetos específicos: las definiciones de esquemas y las definiciones de credenciales. [33]

Una definición de esquema es una definición de un conjunto de tipos de datos y formatos de atributos que pueden ser utilizados para reclamar una credencial. Por ejemplo, un esquema para crear las credenciales de un pasaporte incluiría la definición de varios atributos como: nombre, apellidos, fecha de nacimiento, número de pasaporte, etc. La definición de un esquema puede ser utilizada por una gran variedad de emisores de credenciales y es una manera de asegurar la estandarización entre los emisores.

La posibilidad de que los desarrolladores y los emisores de credenciales publiquen sus propios esquemas y definiciones de credenciales directamente en el libro mayor de Sovrin tiene ventajas significativas. En primer lugar, como no hay una autoridad central, cualquiera puede definir y emitir nuevos tipos de credenciales. Es decir, el intercambio de credenciales está completamente descentralizado. Incluso una organización pequeña, o incluso un individuo, pueden crear nuevos tipos de credenciales que pueden ser emitidas y verificadas instantáneamente por cualquier persona.

Registros de revocación

Existe la posibilidad de tener que revocar una credencial. Un ejemplo común es el permiso de conducir: un usuario puede perder su permiso de conducir si ha cometido varias infracciones graves y reiteradas.

Un sistema SSI que guarda las credenciales (o hashes de credenciales) directamente en la cadena de bloques tiene dificultades para llevar a cabo una revocación. Para evitar este problema, en la red Sovrin no se escriben credenciales (ni siquiera hashes o encriptadas) en el libro mayor. Por lo tanto, ni las credenciales en sí ni las credenciales cifradas se almacenan en el libro de Sovrin. Sólo se emiten y se intercambian fuera del libro entre los agentes de Sovrin y sus respectivas carteras.

Sovrin ha desarrollado una solución descentralizada, asincrónica y privada para la revocación de credenciales. Dicha solución recibe el nombre de “registros de revocación”.

Un registro de revocación es una estructura de datos escrita al libro de Sovrin por el emisor. Hace referencia a la definición de la credencial y contiene un número único y largo llamado “acumulador criptográfico” (cryptographic accumulator). Este número puede ser comprobado instantáneamente por cualquier parte que confíe en él cuando sea necesario asegurarse de que un dato no ha sido revocado por el emisor. El mecanismo es parecido a una especie de función de hash compuesta: el valor del número cambia cuando se añaden o se quitan hash de credenciales válidas en la lista, pero desde el número mismo es imposible saber si alguna credencial concreta está contenida dentro de dicha lista. El único que puede conocer ese dato es el dueño de las credenciales. [33]

Sólo el titular de la credencial puede crear una prueba de no revocación de conocimiento cero, es decir, una prueba de que su credencial pertenece al conjunto de credenciales válidas (sin revelar cuál es). La parte que confía y que necesita saber que una credencial no ha sido revocada puede utilizar esta prueba de no revocación, junto con el acumulador criptográfico que el emisor introdujo previamente en el libro de Sovrin para determinar instantáneamente si la credencial sigue siendo válida.

Cuando un emisor necesita revocar una credencial, lo único que tiene que hacer es "restar" el hash de la credencial del acumulador criptográfico y enviar el nuevo número al ledger de Sovrin. En el momento en que esto ocurre, el poseedor de la credencial ya no podrá producir una prueba válida de no revocación. El orden cronológico del libro mayor garantiza que un verificador sepa siempre que se está utilizando el acumulador más reciente.

Políticas de autorización de agentes

Hay una función más que los titulares de identidades de Sovrin necesitan para proteger su seguridad: la posibilidad de autorizar a su(s) agente(s) en todos sus dispositivos (móvil, PC, ordenador portátil, coche, etc.) a que presenten pruebas de credenciales en su nombre, además de la posibilidad de revocar dicha autorización si el dispositivo se pierde o se ve comprometido.

Para ello, Sovrin ha sido diseñado para dar soporte a políticas de autorización de agentes. Se trata de otro uso especializado de los acumuladores criptográficos y de la criptografía de conocimiento cero en el libro mayor de Sovrin para permitir que el titular de la identidad demuestre a la parte que confía que un agente determinado está autorizado a comunicarse en nombre del titular de la identidad. [33]

Cuando un titular de identidad de Sovrin autoriza a un nuevo agente a actuar en nombre del titular de identidad (por ejemplo, compra un nuevo *smartphone*), el titular de identidad añade una clave de autorización al registro de la política de autorización del agente. Luego, cuando el agente abre una conexión con una parte que confía en él, firma sus mensajes con su clave de autorización, y la parte que confía en él comprueba el registro de la política de autorización del agente para asegurarse de que la clave de autorización no ha sido revocada.

Si el dispositivo se pierde o se ve comprometido, el titular de la identidad desautoriza al agente eliminando la clave de autorización del registro de políticas de autorización del agente.

Almacenamiento de datos personales

El diseño de Sovrin hace que no se guarde ninguna credencial privada en la cadena de bloques. [33] Esto es debido a que, primero, la criptografía se puede romper en algún momento. En segundo lugar, si alguna de las claves privadas se ven comprometidas, o si un emisor o parte confiable con quien han compartido datos encriptados se ve comprometido, un atacante podría ser capaz de recuperar el registro de los datos, ya que en Blockchain nadie puede repudiar ni borrar los datos. Por último, las cadenas de bloques y los libros de contabilidad distribuidos son registros inmutables y permanentes de las transacciones. El almacenamiento de información privada en este medio crea un registro indeleble que puede ser usado para relacionar los datos a

una persona, incluso si la información nunca ha sido descifrada. Por lo que esto supondría un grave problema de privacidad.

Por ejemplo, en el caso de una universidad, esta contaría con un "Sistema de Información de Estudiantes" donde estarían guardados todos los datos sobre los estudiantes: calificaciones, títulos recibidos, etc.

5.2.3 Funcionamiento de Sovrin

En este apartado se va a presentar un resumen del funcionamiento de Sovrin basado en su misma página oficial. [37]

La red Sovrin está formada por nodos de servidores ubicados en todo el mundo, alojados y administrados por un grupo diverso de entidades de confianza llamadas "*Stewards*" o administradores. Cada nodo contiene una copia del libro de contabilidad, es decir, un registro de la información de acceso público necesario para verificar la validez de las credenciales emitidas dentro de la red.

En Sovrin, los *Stewards* realizan unas referencias cruzadas a cada transacción para asegurar la coherencia sobre qué información está escrita en el libro mayor y en qué orden.

Los titulares de las identidades, los emisores de credenciales y las entidades verificadoras acceden a estos servicios en la Red Sovrin mediante unos actores denominados "Agentes". Los agentes se encargan de mantener y procesar las solicitudes en la red Sovrin. Los agentes pueden realizar transacciones de identidad en nombre del propietario de la identidad e intercambiar información directamente con otros agentes mediante conexiones seguras y cifradas entre sí. De esta manera, sólo los identificadores públicos de un emisor se guardan en el libro de contabilidad, mientras que la prueba real de la credencial de un titular de identidad se transmite de forma privada a un validador. Sovrin tiene instrucciones específicas y código desarrollado para la creación de dichos agentes, de modo que diferentes agentes de una variedad de desarrolladores pueden trabajar conjuntamente dentro de la red.

Sovrin permite compartir credenciales digitales fiables. La Red Sovrin está diseñada en base al concepto de "privacidad por diseño", hecho que se manifiesta mediante el uso de identificadores

seudónimos por pares, interacciones entre pares y el permiso de divulgación selectiva de datos personales mediante pruebas de conocimiento cero.

En pocas palabras, cuando un titular de una identidad decide compartir una credencial verificable con una entidad de confianza que utiliza la Red Sovrin, crea una prueba que contiene únicamente la información específica que se solicitó utilizando una combinación de elementos de cualquiera de sus credenciales verificables en su cartera digital. El verificador sólo conoce la información que se ha compartido. El verificador no puede guardar la información captada ni saber de dónde proviene.

Usando la Red Sovrin, cada persona, organización o dispositivo que valide una prueba de un titular de una identidad puede estar completamente seguro de que dicha prueba es exacta y oportuna. Así pues, las empresas también pueden evitar las cargas reglamentarias asociadas al almacenamiento masivo de datos de clientes que podrían ser robados o utilizados indebidamente.

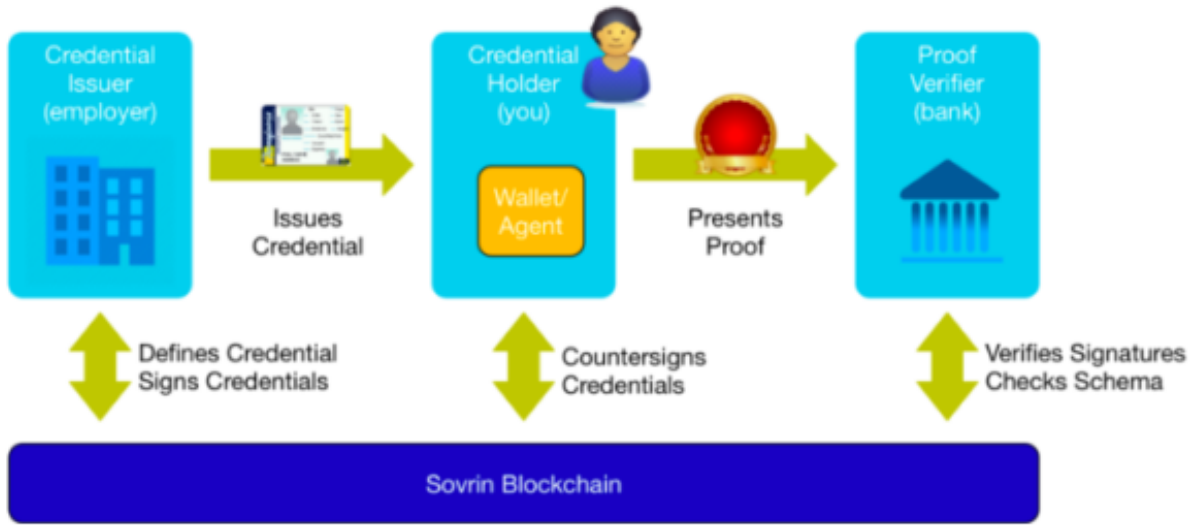

Figura 16 - Funcionamiento de Sovrin [36]

5.3 Hyperledger

5.3.1 Introducción

Hyperledger es un proyecto open-source activo desde el mes de diciembre del año 2015 y creado por la Linux Foundation. Fue ideado con el objetivo de poner en marcha un ecosistema centrado

en crear soluciones Blockchain de código abierto en el ámbito corporativo. Su línea principal de trabajo consiste en intentar aglutinar todos los esfuerzos en el ámbito de la tecnología de las Blockchain permisionadas para desarrollar protocolos y estándares libres y abiertos con el objetivo de crear un ecosistema variado para múltiples usos. Hyperledger se ha convertido en una de las herramientas más destacadas en el fomento del desarrollo de la tecnología Blockchain. Hyperledger proporciona características clave de una cadena de bloques como registro único, inmutabilidad y robustez a la vez que posee características necesarias en un entorno empresarial como la escalabilidad y la privacidad. Este proyecto está pensado para redes Blockchain con un número de participantes conocidos relativamente bajo, lo que le permitiría utilizar algoritmos de consenso mucho más eficientes que los algoritmos empleados por las blockchains públicas como Bitcoin o Ethereum. [38]

En cuanto a los miembros del proyecto Hyperledger, su número va creciendo cada vez más. Entre los miembros más destacados se pueden encontrar algunas de las empresas más influyentes en el mundo de la tecnología como IBM, SAP, Intel o Cisco. También participan algunos bancos y empresas españolas, como por ejemplo el banco BBVA o empresas de investigación como Tecnalia. [39] Cualquier empresa tiene la posibilidad de participar en este proyecto, siendo el único requisito general que las empresas participantes sean miembros de la Linux Foundation. Existen tres tipos de miembros [38]:

1. **Premier Members,** Miembros que realizan una donación anual fija de 250.000 dólares estadounidenses.
2. **General Members.** Miembros que pagan una tarifa en función del tamaño de la empresa.
3. **Associate Members.** Miembros que no tienen que realizar ninguna donación anual, pero han de ser previamente aprobados y deben de ser proyectos de tipo open source, organismos gubernamentales, u organizaciones sin ánimo de lucro.

El Proyecto Hyperledger se compone de tres secciones principales: *frameworks* de código abierto, librerías y herramientas (*tools*), para construir y experimentar con sistemas Blockchain.

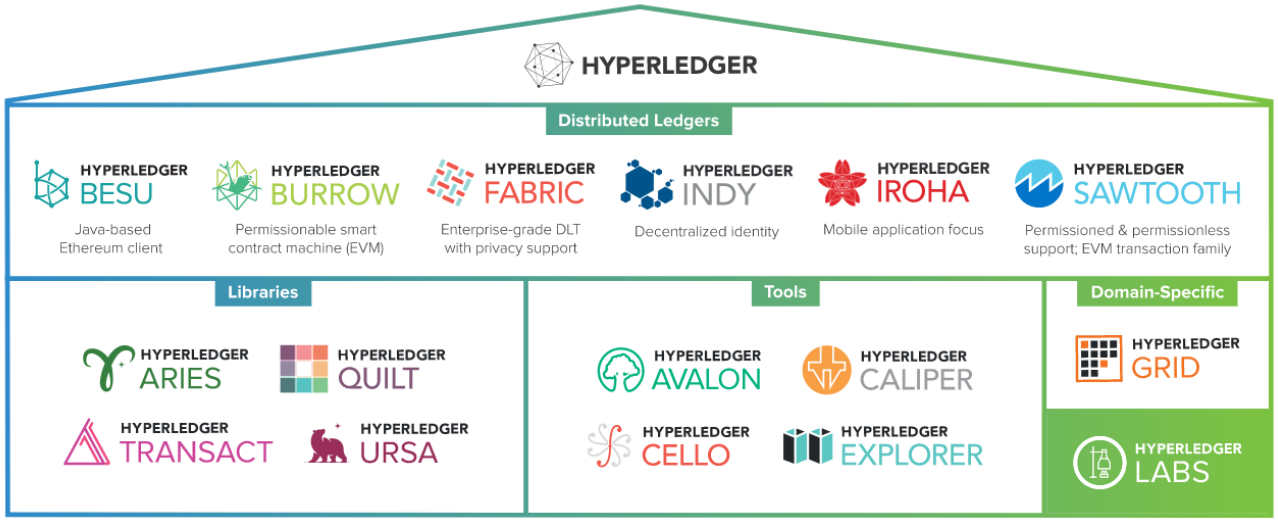

Figura 17 - El ecosistema de Hyperledger [40]

A continuación, se va a proceder a realizar un estudio específico sobre un *framework* concreto de Hyperledger: Indy, ya que se trata de una plataforma cuyo objetivo principal es proponer una solución de identidad digital soberana (SSI).

5.3.2 Hyperledger Indy

Hyperledger Indy es un libro de contabilidad distribuido, diseñado especialmente para la identidad descentralizada. Los desarrolladores pueden utilizar las herramientas y bibliotecas de Hyperledger Indy para crear soluciones de identidad que sean interoperables entre distintas organizaciones y jurisdicciones. Esta interoperabilidad permite a los desarrolladores crear soluciones multisectoriales como *Fintech* y *Healthcare*, que pueden trabajar juntas a la vez que obedecen los estándares regulatorios de cada una. Las especificaciones, terminología y patrones de diseño de Indy son de código abierto, facilitando así el desarrollo de soluciones de identidad descentralizadas. [41]

El código de Indy se basa en el código liberado de Sovrin, la solución que se ha analizado previamente en este documento (apartado 5.2). Por lo tanto, el funcionamiento y las características de Indy resultan ser idénticas a las de Sovrin.

A continuación, se van a explicar brevemente las principales características de Hyperledger Indy.

Roles

Dentro de Hyperledger Indy existen varios roles. Los roles más importantes son los *Trustees*, que equivaldrían a un consejo de administración del sistema. Los *Trustees* nombran a los Administradores que dirigen los dos tipos de nodos de Indy. (Hyperledger, 2018) Los nodos validadores que sirven para actualizar y escribir en la cadena de bloques, y los nodos observadores que otorgan acceso de solo-lectura a la cadena de bloques. Los Administradores designan a los Anclajes de Confianza (*Trust Anchors*), que pueden ser proveedores de identidad, proveedores de atributos o proveedores de servicios. Los usuarios del sistema se denominan Propietarios de Identidad o Custodios de Indentidad. Los primeros controlan su propia identidad mientras que los segundos controlan la identidad de alguien más. Toda la comunicación entre *Trustees*, Propietarios de Identidad y Custodios se hace fuera de la cadena de bloques a través de unos agentes de software. Para ello, se ha creado un sistema descentralizado que emplea la Infraestructura de Clave Pública Descentralizada (DPKI) que no requiere una autoridad central.

Los DIDs

Indy utiliza los DIDs para establecer conexiones entre dos actores con el objetivo de llevar a cabo una comunicación segura. Los DID son identificadores únicos, creados por su propietario, e independientes de cualquier autoridad central.

Cada usuario contará con muchos DIDs, uno para cada relación que tenga en Internet. Además, ambas partes de una relación proveen un DID para hacer posible la comunicación.

Credenciales

Indy incluye un estándar emergente del W3C llamado Credenciales Verificables (VC) que permite proporcionar atributos de identidad de manera fiable y segura.

Las credenciales son certificados (licencias de conducir, pasaportes o títulos universitarios, etc.) otorgados por una autoridad emisora y que se pueden mostrar cuando sea necesario.

Características de Indy

Las características principales de Indy son [42]:

- Ledger distribuido, diseñado específicamente para la identidad descentralizada y soberana.
- Resistente por diseño a la correlación.
- DIDs (Identificadores Descentralizados) globalmente únicos y resolubles (a través de un libro mayor) sin requerir de ninguna autoridad de resolución centralizada.
- Identificadores Paritarios que establecen relaciones seguras, 1:1 entre dos entidades cualesquiera.
- Reclamaciones Verificables: un formato interoperable para el intercambio de atributos y relaciones de identidad digital que actualmente se encuentra en proceso de estandarización en el W3C.
- Pruebas de Conocimiento Cero: mecanismo para probar que algunos o todos los datos de un conjunto de peticiones son verdaderos, sin revelar ninguna información adicional, incluyendo la identidad del procurador.

Repositorios para desarrolladores

Indy cuenta con varios repositorios de código abierto en Github para facilitar la puesta en marcha de soluciones de identidad soberana:

- https://github.com/Hyperledger/indy-node. Esta base de código contiene toda la funcionalidad para ejecutar nodos (validadores y/u observadores) que proporcionan un ecosistema de identidad soberana sobre un libro mayor distribuido. Es el proyecto central de Indy.
- https://github.com/Hyperledger/indy-plenum. Plenum es el núcleo de la tecnología del libro mayor distribuido dentro de Hyperledger Indy.
- https://github.com/Hyperledger/indy-sdk. Este es el SDK oficial de Hyperledger Indy, que proporciona la base del libro mayor distribuido para la identidad soberana.
- https://github.com/hyperledger/aries. El antiguo repositorio indy-agent que contenía la base de código para los agentes de confianza de un sistema SSI ha sido absorbido por el proyecto Hyperledger Aries. Aries es una infraestructura de herramientas que permite el intercambio de datos basados en cadenas de bloques, implementa la mensajería P2P en

varios escenarios y facilita la interacción entre diferentes cadenas de bloques y otras tecnologías de libro mayor distribuido (DLTs).

5.4 uPort

5.4.1 Introducción

uPort es un sistema de identidad descentralizada (soberana) basado en la cadena de bloques de la plataforma Ethereum. uPort también contiene una colección de herramientas y protocolos enfocados al desarrollo de aplicaciones descentralizadas. Además, está construido sobre estándares abiertos y bibliotecas de código abierto.

Las identidades de uPort pueden ser creadas y manejadas a través de los clientes de uPort, como, por ejemplo, su aplicación móvil. Las identidades son propiedad del creador en su totalidad y no dependen de terceros centralizados para su creación, control o validación. [43]

Características principales de uPort: [44]

- Identidad radicada en la cadena de bloques de Ethereum
- Inicio de sesión sin contraseña (Código QR)
- Gestión simplificada de claves
- Vinculación de firmas digitales
- Sistema de reputación del usuario
- Verificación de la identidad
- Comunicación entre el navegador y la aplicación móvil (Ethereum *DApps*[1])
- Soporte para cadena de bloques privada
- El almacenamiento de los datos del usuario se mantiene fuera de la cadena para conservar la privacidad. En su lugar, se debe solicitar la información privada directamente al usuario.

[1] *Una DApp es una aplicación basada en Blockchain.*

En el Anexo II “Información adicional”, se describe paso a paso todo el proceso necesario para realizar alguna acción en uPort (firma de transacciones, recuperación de claves, etc.).

5.4.2 Clientes de uPort

Aplicaciones que pueden interactuar con la plataforma del uPort [44]:

- **Aplicación móvil uPort.** Cartera móvil segura que le permite al usuario crear su identidad, gestionar sus datos y gestionar las solicitudes.
- **uPort App Manager.** Cliente web para que los desarrolladores puedan gestionar la identidad uPort de sus aplicaciones.
- **Cliente JS de uPort.** Cliente JavaScript diseñado para que los desarrolladores interactúen con la plataforma uPort. Mediante esta herramienta se pueden crear identidades, manejar mensajes o ejecutar firmas.

5.4.3 Arquitectura

La arquitectura del uPort incluye tres elementos principales [44]:

1. **Contratos inteligentes** que certifican la identidad del usuario y que contienen el sistema que permite al usuario recuperar su identidad en caso de pérdida o robo de su dispositivo móvil.
2. Una **aplicación móvil** que contiene las claves del usuario y le permite comunicarse con el contrato inteligente (transacción de firma). La clave se mantiene almacenado de forma segura dentro del dispositivo del usuario y se accede a ella mediante la autenticación biométrica local siempre y cuando la clave se utilice para firmar. La clave permanece en el dispositivo y no hay forma de exportarla fuera de este.
3. Las **librerías para desarrolladores** son el mecanismo mediante el cual los desarrolladores de aplicaciones de terceros integrarían el soporte para uPort en sus aplicaciones.

1. Contratos inteligentes

El identificador de uPort es una cadena hexadecimal de 20 bytes que actúa como un identificador global único y permanente. Este identificador se define como la dirección de un contrato inteligente de Ethereum, también conocido como **contrato proxy**.

El contrato proxy sirve para retransmitir las transacciones y es a través de este mecanismo como la identidad interactúa con otros contratos inteligentes en la cadena de bloques Ethereum. Cuando el usuario quiere identificarse e interactuar con un contrato inteligente de una aplicación cualquiera, envía una transacción a través del contrato proxy, mediante un **contrato controlador** que contiene la lógica de acceso. Después, el contrato proxy reenvía esta transacción al **contrato inteligente de la aplicación.** [44]

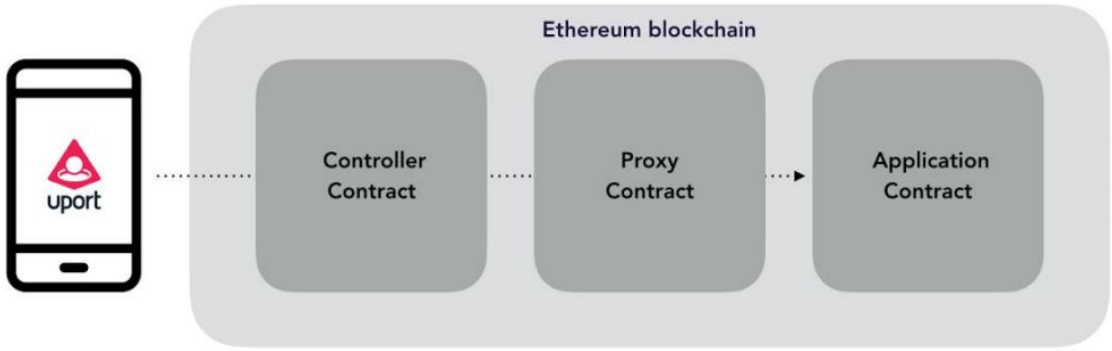

Figura 18 - Arquitectura de uPort [44]

Esta arquitectura le permite a la aplicación ver la dirección del contrato proxy como la entidad que interactúa. Así pues, el contrato proxy crea una capa de in-dirección entre la clave privada del usuario (almacenada en su dispositivo móvil) y el contrato inteligente de la aplicación.

El propósito de contar con un contrato proxy como identificador central es que así se le permite al usuario reemplazar su clave privada mientras mantiene un identificador permanente. Si el identificador del usuario fuera la clave pública correspondiente a su clave privada, perdería el control sobre su identificador al perder el dispositivo donde se guarda la clave privada. [44]

¿Qué ocurre si el usuario pierde o destruye su dispositivo móvil?

De manera similar a lo que ocurre en el sistema de Sovrin, en caso de pérdida de dispositivo, uPort permite al usuario asociar una nueva clave pública a través de la intervención de un quórum de personas de confianza. Estos agentes de confianza (conocidos en uPort como "delegados de recuperación") son definidos por el usuario y pueden ser individuos, como amigos y familiares, o instituciones. [44]

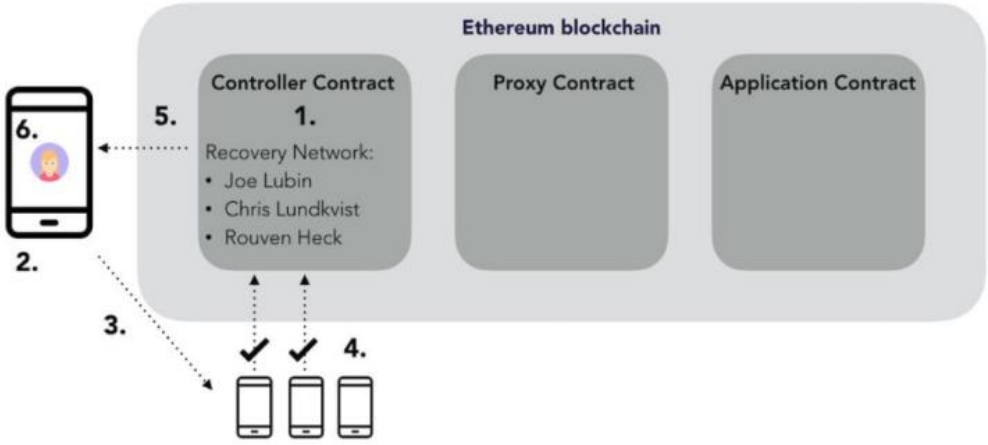

Figura 19 - Esquema de recuperación de identidad en uPort [44]

En resumen, un usuario que ha perdido el acceso a su dispositivo debe realizar los siguientes pasos [44]:

1. El usuario tiene una red de recuperación guardada en su contrato de controlador (direcciones de recuperación)
2. Consigue un nuevo teléfono
3. Le comunica a su red de recuperación sobre su nueva clave de dispositivo público
4. Los contratos de recuperación le confirman la nueva clave de dispositivo al contrato controlador
5. El contrato controlador actualiza la clave pública del usuario
6. Se recupera la identidad

2. La aplicación móvil

Pasos para interactuar con una DApp [44]:

1. **"Conectar".** Proporcionarle el identificador de uPort a la DApp. La versión de escritorio de estas aplicaciones muestra un código QR, y al escanear dicho código QR desde la versión móvil de la aplicación, el usuario conseguirá el acceso al sistema.
2. **"Verificar/validar/autorizar" una interacción.** Firmar una transacción con la clave privada. Cuando el usuario necesita confirmar una interacción con la cadena de bloques (es decir, firmar una transacción) se muestra otro código QR. El usuario escanea dicho código y a continuación se le muestra una pantalla de confirmación en la que puede confirmar la interacción mediante su huella dactilar.

3. Librerías de uPort

uPort posee una librería llamada "uport-connect" en el lado del cliente que permite interactuar con la identidad de un usuario de uPort a través de un cliente de la plataforma, principalmente la aplicación móvil. Dicha librería maneja el canal de comunicación entre la aplicación y un cliente uPort, que puede variar dependiendo del entorno en el que se ejecute su aplicación. A través de este canal de comunicación se le pueden enviar solicitudes de datos a un usuario, compartir credenciales y generar transacciones para ser firmadas. Esta librería ofrece una implementación *"quick-start"* por defecto para integrarse con uPort. Si dicha solución no cumple con las necesidades del usuario, este también puede usar dos librerías alternativas: "uport-tranports" y "uport-credentials". [43]

uport-transports es una colección de funciones acopladas llamadas "transportes" y funciones de utilidad empleadas para establecer canales de comunicación entre una aplicación y un cliente uPort. Los transportes son funciones que absorben mensajes de solicitud y parámetros de transporte adicionales, y luego los envían a un cliente uPort. Algunos transportes también gestionan la recepción de una respuesta a una solicitud determinada. Muchas de estas funciones pueden combinarse para crear transportes específicos para un caso de uso concreto. [45]

uport-credentials simplifica el proceso de creación de identidades dentro de las aplicaciones JavaScript; además, ayuda a que las aplicaciones verifiquen y firmen los datos de manera más

sencilla (firmados por otras identidades para facilitar la comunicación segura entre las partes). Estos datos se representan en forma de *tokens* web JSON firmados (JWT), que cuentan con campos específicos diseñados para su uso con clientes de uPort. [46]

5.4.4 Servicios centrales

uPort hace uso de tres servicios centrales para llevar a cabo algunas tareas imprescindibles para el correcto funcionamiento del sistema. Dichos servicios son [44]:

- **Chasqui - servidor de mensajería.** Chasqui se encarga de la comunicación desde el frontend de una aplicación descentralizada de escritorio hacia (y desde) la aplicación móvil.
- **Sensui - servidor de abastecimiento.** Sensui ayuda a los nuevos usuarios de Ethereum a superar el obstáculo inicial de tener que comprar *ethers* (la criptomoneda de Ethereum) para pagar las cuotas necesarias para utilizar la red. Sensui realiza el pago de las tarifas de *gas*[2] para el usuario, permitiéndole crear un nuevo uPort y ponerlo en funcionamiento de manera inmediata.
- **Infura Ethereum RPC.** El proveedor de endpoints RPC Ethereum, Infura, permite que uPort se comunique con la red Ethereum a través de la interfaz RPC estándar que proporciona Infura. Este sistema hace posible el uso de la aplicación móvil de uPort hasta que existan clientes móviles ligeros y avanzados disponibles para Ethereum.

[2] *El gas es el coste que tiene el realizar una operación o un conjunto de operaciones en la red Ethereum.* [47]

5.4.5 Debilidades y futuras soluciones

Según la *review* oficial de uPort publicada en "Medium" [44], el sistema de uPort aún tiene algunas debilidades y aspectos que deben ser mejorados. Estos aspectos son:

- Todos los delegados de recuperación establecidos por un usuario son visibles de forma pública en la cadena de bloques. Esto podría suponer un riesgo de seguridad, ya que un

atacante podría decidir atacar a los delegados de un usuario para comprometer su identidad.

- Los datos almacenados temporalmente en el servidor de Chasqui no están encriptados, lo que puede suponer un problema de privacidad. Próximamente se hará que todas las comunicaciones de este servidor estén encriptadas de extremo a extremo.
- En un futuro, el servidor de Sensui empleará una arquitectura más sofisticada para realizar el pago de las cuotas de usuario, lo que implicará una lógica de contrato inteligente incorporada en el contrato controlador.

5.5 Comparativa de las soluciones analizadas

En esta sección se presenta una tabla (ver Tabla 3) con una comparativa entre las principales características de las soluciones analizadas: Alastria ID, Sovrin / Hyperledger Indy y uPort.

Explicación de las filas de la tabla:

Blockchain permisionada / no-permisionada - El tipo de Blockchain sobre el cual está basado el sistema de identidad soberana (pública o privada).

Minado (prueba de trabajo) - Si es necesario un algoritmo de consenso para validar los datos de la Blockchain. Este algoritmo suele ser necesario en el caso de las blockchains no-permisionadas (públicas).

Gestión de claves - Cómo se gestionan las claves privadas de los usuarios.

Importación de datos de identidad - Si se le permite al usuario crear atributos de identidad en el sistema. Esta característica suele requerir algún método de verificación y confianza.

Divulgación selectiva - Esta característica implica el hecho de que el usuario pueda compartir con terceras partes solo los datos que considere convenientes.

Almacenamiento de los datos - Cómo se almacenan los datos personales de los usuarios (dentro o fuera de la Blockchain, dónde…).

Confianza requerida - Si se requiere la presencia de algún organismo de confianza. Las Blockchain públicas no requieren ningún tipo de confianza debido a que poseen un algoritmo de consenso (prueba de trabajo).

Smart Contracts - Si la solución permite la ejecución de contratos inteligentes. Las soluciones SSI basadas en Ethereum suelen poseer esta característica.

Tabla 3 - Comparación de características de las soluciones SSI

	Alastria ID	**Sovrin / Indy**	**uPort**
Blockchain permisionada / no-permisionada	Semipública - permisionada	Permisionada	Pública
Minado (prueba de trabajo)	No	No	Sí
Gestión de claves	El dispositivo del usuario y *DPKI* [5]	DPKI	El dispositivo del usuario y DPKI
Importación de datos de identidad	?	Sí	Sí
Divulgación selectiva	Sí	Sí	Sí
Almacenamiento de los datos	Dentro y Fuera de Blockchain	Dentro y fuera de la Blockchain	Fuera de la Blockchain
Confianza requerida	Sí	Sí	No
Smart Contracts	Sí	No	Sí

Dada la tabla anterior, se la llegado a la conclusión de que Sovrin / Hyperledger Indy es la tecnología más prometedora gracias a su gestión de claves, la no-necesidad de implementar un algoritmo de consenso y el soporte a la importación de datos de identidad.

Alastria ID también es una propuesta interesante, ya que a diferencia de Sovrin / Indy, también permite la creación de contratos inteligentes. Sin embargo, esta solución está limitada actualmente solo al mercado español y no cuenta con una comunidad tan amplia ni un estado de desarrollo tan avanzado como Sovrin / Indy.

[3] *DPKI - Infraestructura descentralizada de claves públicas.*

6. Diseño y desarrollo de un sistema SSI

Este proyecto tiene como objetivo principal la puesta en marcha de una solución SSI basada en Blockchain para el intercambio de expedientes médicos teniendo en cuenta el GDPR.

Sin embargo, antes se va a proceder a diseñar y poner en marcha una solución SSI con un caso de uso diferente, un caso de uso más común y más básico. Posteriormente dicho sistema se adaptará para cumplir con el objetivo principal descrito anteriormente.

El sistema recibirá el nombre de “UOC-ID”.

6.1 Diseño

6.1.1 Decisiones de diseño

Tal y como se ha especificado antes, la solución SSI a diseñar, desarrollar y probar tendrá que estar basada en la tecnología Blockchain. El empleo de una la cadena de bloques es fundamental ya que esta tecnología es la que mejor asegura la soberanía de los datos, emplea una encriptación muy robusta y asegura la confianza dentro del sistema.

Es necesario establecer sobre qué base será construido el sistema, ya que la puesta en marcha de un sistema SSI desde cero es inviable dado el escaso tiempo disponible y la falta de un equipo de trabajo más amplio.

Tipo de Blockchain a emplear (pública o privada)

Para la creación y la actualización de bloques, los nodos de la red deben tener cierto control. Una cadena de bloques pública no asegura la confianza en un sistema, por lo que sería necesario implementar un algoritmo de consenso muy costoso. Además, las Blockchain permisionadas son mucho más escalables que las cadenas públicas y su crecimiento puede ser controlado. [58] Por último, dado que el objetivo final de este sistema es ser adaptado para el ámbito de la medicina, no sería adecuado contar con una Blockchain pública, abierta a todo el mundo.

Uso de Hyperledger Indy

Existen varias cadenas de bloques permisionadas que podrían servir. Se ha elegido Hyperledger Indy debido a que es una solución diseñada específicamente para el desarrollo de sistemas SSI, cumple con el principio de "privacidad por diseño", es de tipo permisionado y cuenta con un amplio apoyo por parte de grandes organizaciones internacionales como Sovrin y Linux Foundation, además de una amplia comunidad de desarrolladores que está empeñada en mejorar cada vez más su código fuente.

Almacenamiento de datos personales

Tal y como se ha especificado en el apartado 5.2.2 "Aspectos técnicos avanzados de Sovrin", el diseño de Sovrin-Indy hace que no se guarde ninguna credencial privada en la cadena de bloques. Esto es debido a que, primero, la criptografía se puede romper en algún momento. En segundo lugar, si alguna de las claves privadas se ven comprometidas, o si un emisor o parte confiable con quien han compartido datos encriptados se ve comprometido, un atacante podría ser capaz de recuperar el registro de los datos, ya que en Blockchain nadie puede repudiar ni borrar los datos. Por último, las cadenas de bloques y los libros de contabilidad distribuidos son registros inmutables y permanentes de las transacciones. El almacenamiento de información privada en este medio crea un registro indeleble que puede ser usado para relacionar los datos a una persona, incluso si la información nunca ha sido descifrada. Por lo que esto supondría un grave problema de privacidad.

Por lo tanto, en el sistema a poner en marcha tanto aquí en el capítulo 6 como en el capítulo 7 no se van a guardar datos personales en la cadena de bloques. Dichos datos serán definidos manualmente para poder llevar a cabo la demostración del caso de uso definido (apartado 6.1.2), pero en un escenario real, los datos deben proceder de una aplicación empresarial de algún tipo. Por ejemplo, en el caso de una universidad, esta contaría con un "Sistema de Información de Estudiantes" donde estarían guardados todos los datos sobre los estudiantes: calificaciones, títulos recibidos, etc.

Interfaz web (gráfica) del sistema SSI

El sistema SSI debe contar con una interfaz web para facilitar su uso y las pruebas que se realizarán sobre el entorno. Para ello se ha elegido realizar una adaptación de la interfaz construida con NodeJS que viene por defecto en Indy para adaptarla al sistema que se va a desarrollar.

6.1.2 Caso de uso

El caso de uso a poner en marcha es el siguiente:

Una **persona (Denis)** desea adquirir un coche, pero el **concesionario (Ford)** le exige presentar una prueba que demuestre que posee el permiso de conducir. Para ello, Denis tiene que obtener la prueba de que posee dicho documento desde la **DGT**, que ha también ha implementado recientemente la gestión de los permisos de conducir mediante un sistema de identidad soberana.

Por lo tanto, el diseño del sistema en forma de diagrama quedaría de la siguiente manera:

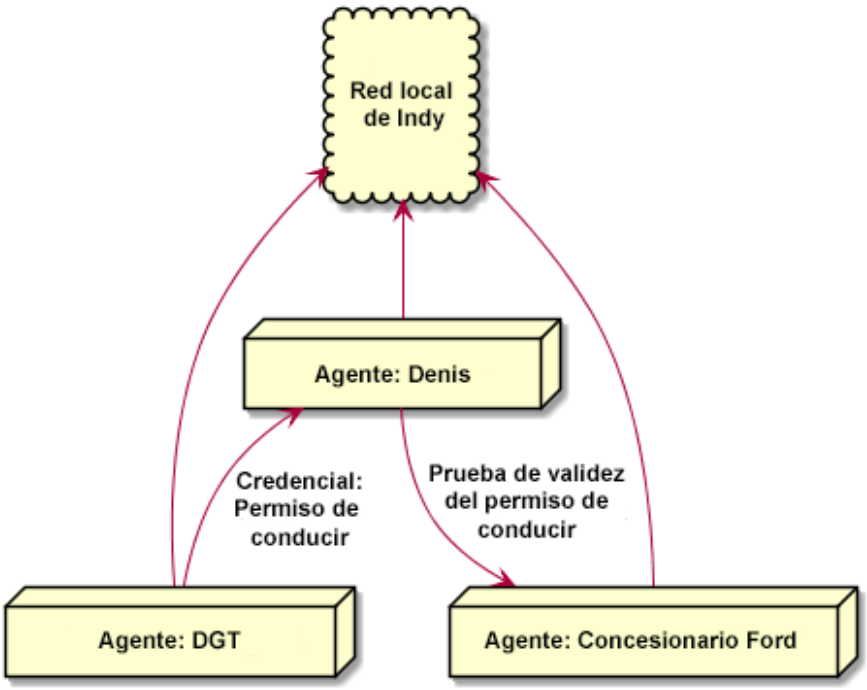

Figura 20 - Diagrama del sistema

6.1.3 Herramientas técnicas para el desarrollo

A continuación, se van a exponer todas las herramientas necesarias para poder llevar a cabo este proyecto:

- **El SDK y la librería libindy.** El SDK ofrece todo lo necesario para poner en marcha aplicaciones personalizadas basadas en Indy. Se trata de un código fuente preparado para dicho cometido. URL: https://github.com/hyperledger/indy-sdk/blob/master/docs/getting-started/indy-walkthrough.md (2020)
  Consulte el ANEXO I: GUÍA DE INSTALACIÓN para información acerca de cómo instalar libindy en Ubuntu 18.04 LTS.
- **Entorno de desarrollo integrado (IDE).** La librería libindy ofrece distintos envoltorios (*wrappers*) para facilitar la implementación de aplicaciones basadas en Indy en varios lenguajes de programación. Indy SDK proporciona envoltorios libindy para los siguientes lenguajes y plataformas de programación: Java, Python, iOS, NodeJS, .NET y Rust.
  En el caso de esta aplicación se ha elegido el lenguaje NodeJS: https://nodejs.org/es/
- **Entorno para el *pool* de nodos.** Para crear un conjunto de nodos de forma local, *indy-nodes* ofrece la posibilidad de utilizar Docker (https://www.docker.com/). Para utilizar Docker en Windows se requiere la versión Pro de Windows 10. Por lo tanto, al no disponer de dicha versión, se va a utilizar una máquina virtual con Ubuntu 18.04 LTS (Linux). Dicha máquina virtual será creada mediante el software VMware (licencia gratuita para uso no-comercial, https://www.vmware.com/)
  Para más información acerca de cómo instalar Ubuntu 18.04 y Docker, consulte el ANEXO I: GUÍA DE INSTALACIÓN.

6.1.4 Diseño de la interfaz web

Tal y como se ha comentado en el apartado 6.1.1 - “Interfaz web (gráfica) del sistema SSI”, el sistema SSI debe contar con una interfaz web para facilitar su uso y las pruebas que se realizarán sobre el entorno. Para ello se ha elegido realizar una adaptación de la interfaz construida con NodeJS que viene por defecto en Indy para adaptarla al sistema que se va a desarrollar.

Dicha interfaz web cuenta con dos partes principales:

1. **Una pantalla de logueo** para los diferentes actores involucrado: Denis, la DGT y el concesionario de coches Ford. Ahí cada usuario debería introducir un nombre de usuario y una contraseña.
2. **Una pantalla (principal) de gestión de la identidad** para cada uno de los actores involucrados.

   Dicha pantalla contaría con los siguientes apartados:

   - **Relaciones.** En este apartado el usuario podrá visualizar las relaciones que tiene con otros actores. La información que podrá ver sobre cada uno de sus contactos dependerá del consentimiento que estos le han dado (o no) para establecer una relación.

     Además, en esta sección también se podrán crear nuevas relaciones a través del Endpoint DID del usuario con el que se desee contactar.
   - **Credenciales.** En esta sección el usuario podrá visualizar sus credenciales: DNI, permiso de conducir, certificados académicos, etc. Es decir, todos los certificados que le pertenecen y que ha solicitado previamente a sus instituciones correspondientes.
   - **Petición de pruebas (proof request).** En esta sección un usuario u organización puede crear una petición de pruebas para poder verificar la autenticidad de una credencial. Por ejemplo, una empresa que desee contratar a Denis como responsable de ciberseguridad puede crear un *proof request* y mandárselo a la UOC para probar que el título de Máster en Seguridad de las TIC que Denis ha presentado es válido.
   - **Emisión.** En este apartado, una organización puede emitir credenciales para los usuarios. Por ejemplo, la UOC puede emitir la credencial del título de Máster en Seguridad de las TIC para Denis.
   - **Mensajes.** Este apartado sirve para la comunicación entre dos actores. Es decir, si Denis quiere establecer una relación con la UOC, entonces le mandará una petición a dicha institución desde el apartado de Relaciones utilizando su Endpoint DID. Por lo tanto, a la UOC le aparecerá un mensaje con dicha petición de relación por parte de Denis, y esta podrá aceptarla o rechazarla.

6.1.5 Componentes del sistema

Los DIDs

Indy utiliza los DIDs para establecer conexiones entre dos actores con el objetivo de llevar a cabo una comunicación segura. Los DID son identificadores únicos, creados por su propietario, e independientes de cualquier autoridad central.

Cada usuario contará con muchos DIDs, uno para cada relación que tenga en Internet. Además, ambas partes de una relación proveen un DID para hacer posible la comunicación.

Las siguientes dos figuras (24 y 25) muestran los DIDs de una manera más visual:

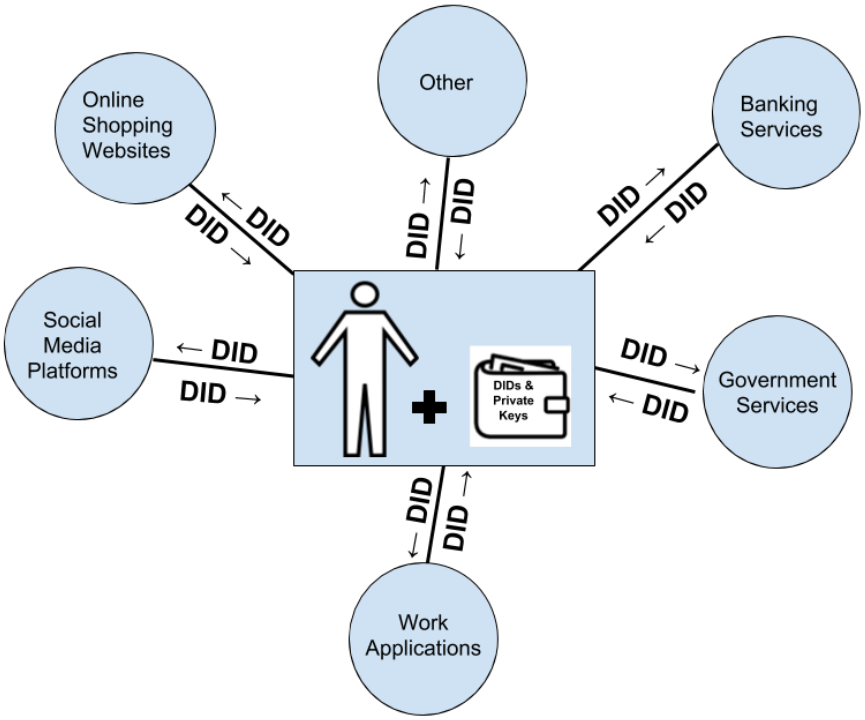

Figura 21 - El usuario y sus DIDs. Fuente: Hyperledger Indy - Licencia CC BY 4.0

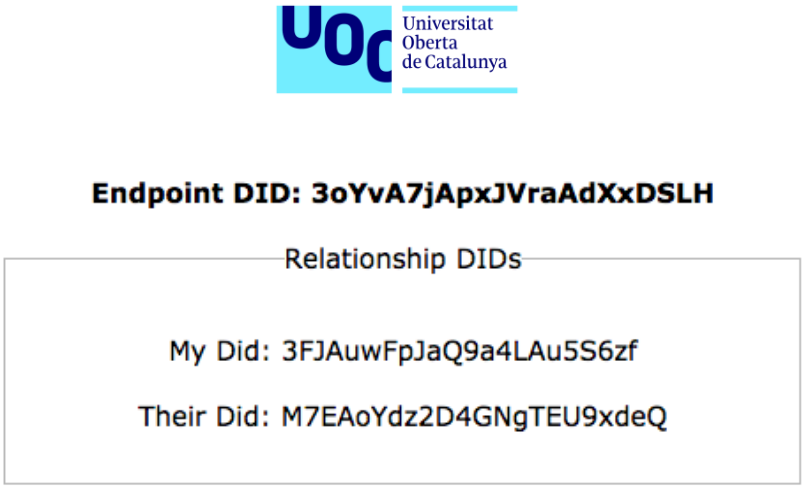

Figura 22 - DIDs. Fuente: Hyperledger Indy - Licencia CC BY 4.0

Agentes y carteras (*wallets*)

Indy utiliza el término "agente" (*agent*) para referirse al software que interactúa con otras identidades (a través de los DIDs), y el término "Cartera" (*wallet*) como un almacén de datos para los DIDs y la información relacionada (incluyendo las claves privadas). Por ejemplo, una persona puede tener una aplicación *Agent* en su dispositivo móvil, mientras que una organización puede tener un *Agent* empresarial ejecutándose en un servidor en la nube. Todos los agentes tienen una cartera segura para almacenar los datos de identidad.

Credenciales

Indy incluye un estándar emergente del W3C llamado Credenciales Verificables (VC) que permite proporcionar atributos de identidad de manera fiable y segura.

Las credenciales son certificados (licencias de conducir, pasaportes o títulos universitarios, etc.) otorgados por una autoridad emisora y que se pueden mostrar cuando sea necesario.

6.2 Puesta en marcha y testeo

Para poner en marcha la red virtual con Docker hay que ejecutar los comandos mostrados a continuación. Para evitar errores por "falta de permisos", se recomiendo ejecutarlos como administrador, utilizando el prefijo "sudo".

$ sudo ./manage build

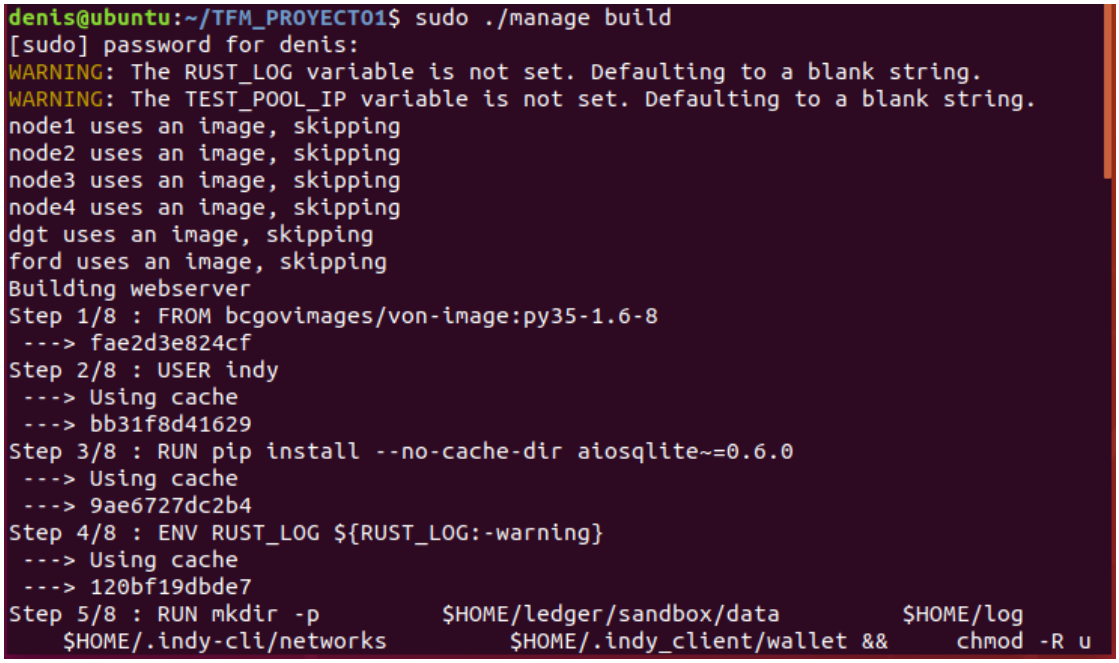

Figura 23 - Puesta en marcha de la red con Docker (1)

$ sudo ./manage up

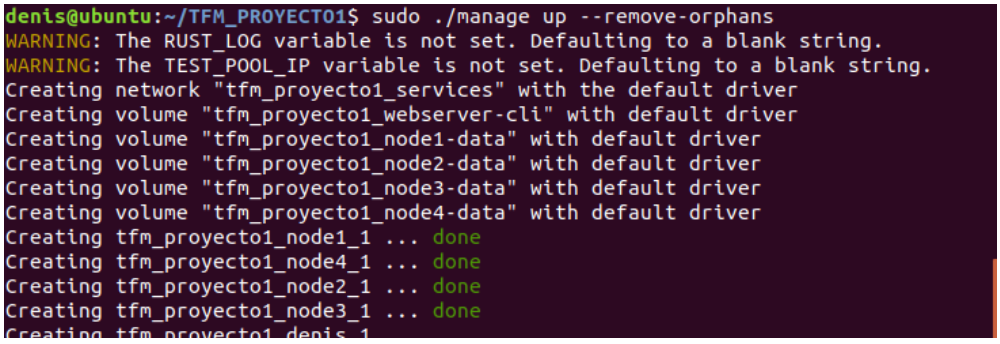

Figura 24 - Puesta en marcha de la red con Docker (2)

Ahora, se puede acceder a cada uno de los agentes desde el navegador web, en este caso, Mozilla Firefox.

Los puertos donde se ejecuta cada agente son:

- Denis - **puerto 3000**
- DGT - **puerto 3002**
- Concesionario Ford - **puerto 3003**

Al entrar en cualquiera de los puertos expuestos anteriormente, se muestra una pantalla de logueo:

Figura 25 - Pantalla de logueo

Tal y como se ha definido en el fichero ***docker-compose.yaml***, las credenciales de acceso para cada agente son:

- **Denis:**
  - Usuario: denis
  - Contraseña: 123
- **DGT:**
  - Usuario: dgt
  - Contraseña: 123
- **Concesionario Ford:**
  - Usuario: ford
  - Contraseña: 123

Una vez dentro del sistema (en este caso, logueado como Denis), aparece la siguiente página con los apartados mencionados en 6.1.4 - Diseño de la interfaz web:

Figura 26 - Pantalla principal

Actualmente el usuario Denis no tiene ninguna relación. Para poder comprar el coche deseado, Denis necesita obtener la credencial que demuestre que posee el permiso de conducir clase B. Para ello, necesita establecer una conexión con la DGT. Esto se haría pinchando el botón de "Mandar solicitud de conexión".

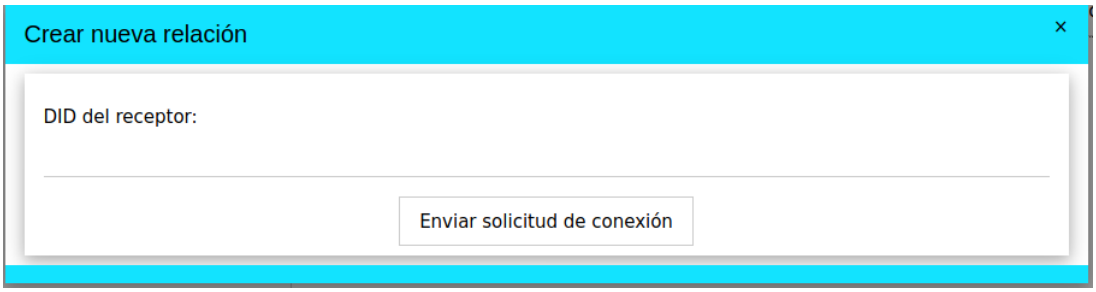

Figura 27 - Enviar solicitud de conexión

En la ventana de "Crear nueva relación" se necesita pegar el DID del otro actor, en este caso la DGT. El DID de la DGT aparece en la parte de debajo de su interfaz:

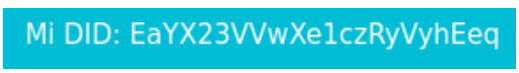

Figura 28 - DID de la DGT

Ahora Denis ya puede mandarle a la DGT una solicitud de conexión:

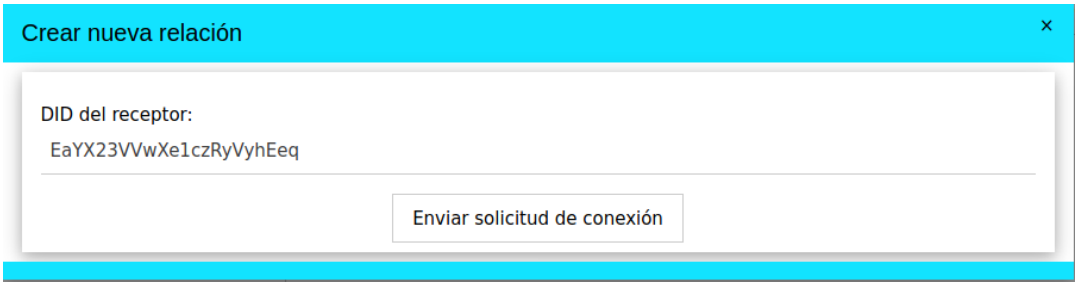

Figura 29 - Enviar solicitud de conexión con el DID de la DGT

A la DGT le aparecerá un mensaje con la solicitud de conexión que Denis ha mandado:

Figura 30 - Solicitud de conexión

Una vez aceptada la conexión **por parte de ambas partes**, a cada actor le aparecerá información sobre el otro en el apartado de "Relaciones". En este caso, la conexión solo requiere compartir un dato personal: el nombre.

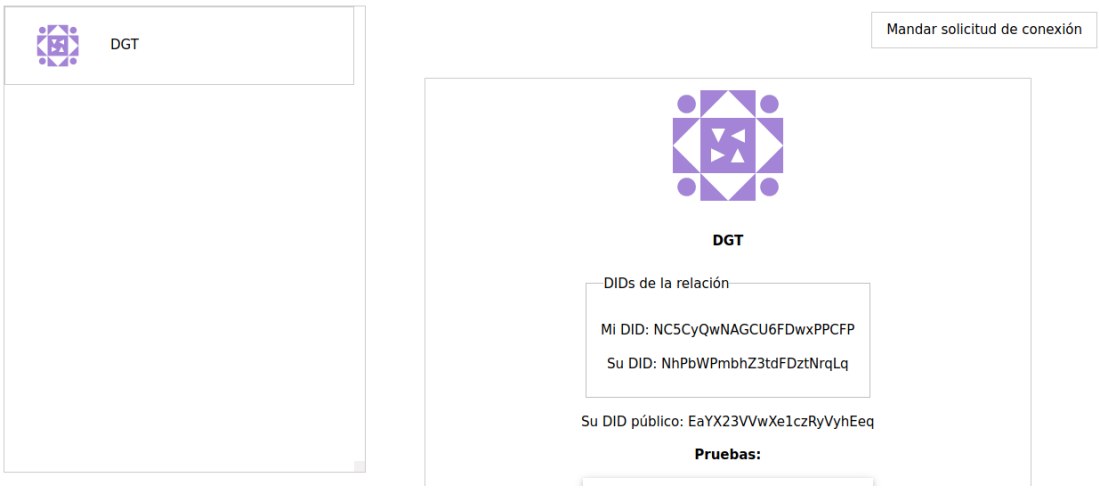

Figura 31 - Pantalla de "Relaciones" de Denis

Ahora, la DGT necesita emitir la credencial que pruebe que Denis posee el carnet de conducir. Para ello, en el apartado de “Emisión” puede crear la credencial del carnet de conducir y mandársela a Denis.

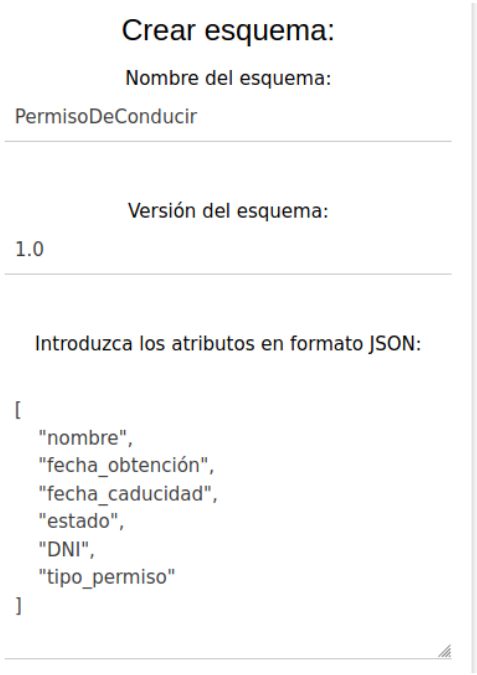

Figura 32 - Paso 1: Crear esquema

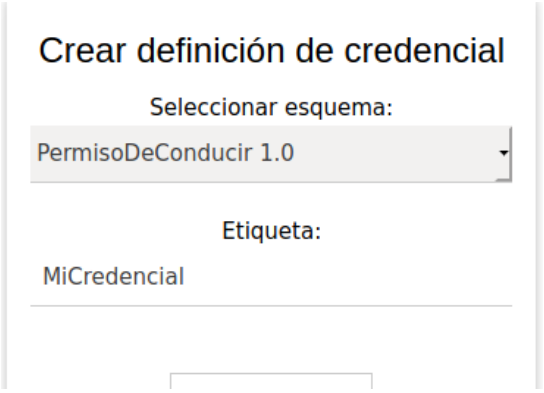

Figura 33 - Paso 2: Crear credencial

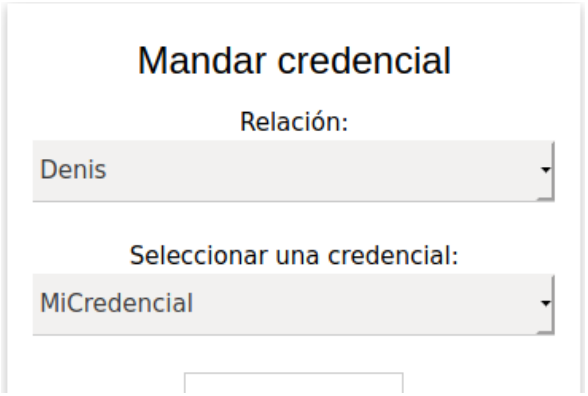

Figura 34 - Paso 3: Mandar credencial

Ahora a Denis le aparecerá un mensaje con la credencial que la DGT acaba de mandar:

Figura 35 - Petición de credencial

Si Denis acepta la credencial, ahora le aparecerá dicha credencial en el apartado de credenciales, junto a su “Government-ID”, credencial que viene por defecto en Hyperledger Indy.

Figura 36 - Página de credenciales de Denis

Ahora que Denis ya tiene su permiso de conducir, va a crear una relación con el concesionario de Ford, y este le mandará a Denis una petición de pruebas para que este demuestre que tiene un permiso de conducir, reclamando tres de sus atributos: estado, DNI y tipo_permiso. Para crear una petición de pruebas del carnet de conducir hace falta copiar la lista de atributos en formato JSON de la cuenta de la DGT, ya que fue esta la que creó la credencial. Esta característica podría ser mejorada implementando un generador de peticiones de pruebas.

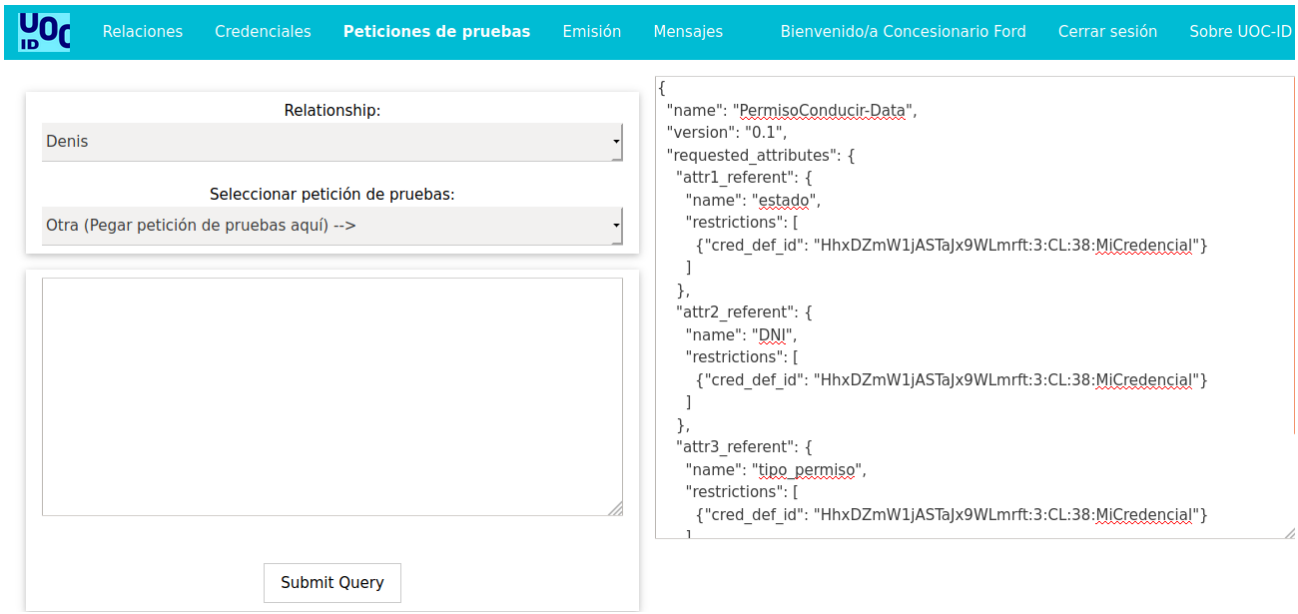

Figura 37 - Crear petición de pruebas para el carnet de conducir

Ahora Denis tendrá que dar su consentimiento para proporcionar los atributos de la credencial (el carnet de conducir) necesarios para la Prueba.

Figura 38 - Proporcional atributos necesarios para la prueba

Como se puede ver, Denis solo va a compartir algunos atributos necesarios del carnet de conducir, y no todos. Esto es una ventaja importante de la identidad soberana.

Ahora, en la relación de Ford con Denis, el concesionario ya puede validar los atributos necesarios para demostrar que Denis tiene un permiso de conducir asociado a su DNI, válido y de tipo B (para conducir un turismo). **Denis ya puede adquirir un coche Ford.**

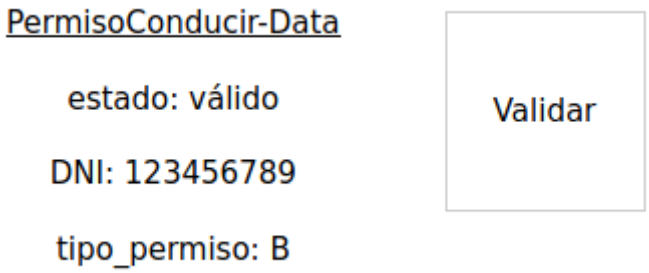

Figura 39 - Validar atributos carnet de conducir (1)

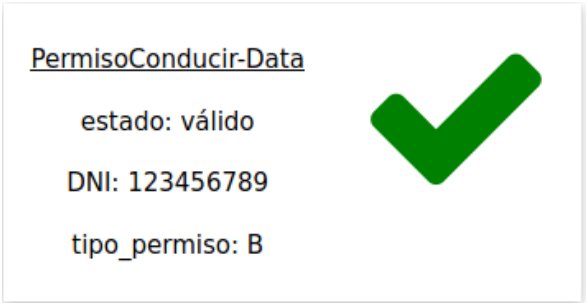

Figura 40 - Validar atributos carnet de conducir (2)

Finalmente, para cerrar el sistema, hace falta introducir el siguiente comando:

$ sudo ./manage down

```
denis@ubuntu:~/TFM_PROYECTO1$ sudo ./manage down
WARNING: The RUST_LOG variable is not set. Defaulting to a blank string.
WARNING: The TEST_POOL_IP variable is not set. Defaulting to a blank string.
Stopping tfm_proyecto1_dgt_1       ... done
Stopping tfm_proyecto1_ford_1      ... done
Stopping tfm_proyecto1_denis_1     ...
Stopping tfm_proyecto1_webserver_1 ... done
Stopping tfm_proyecto1_node4_1     ...
Stopping tfm_proyecto1_node2_1     ...
Stopping tfm_proyecto1_node1_1     ...
Stopping tfm_proyecto1_node3_1     ...
```

Figura 41 - Apagar el sistema

7. Sistema SSI para el intercambio de expedientes médicos

6.1 Introducción

El historial médico electrónico es un tipo de información privada crítica y altamente necesaria para el diagnóstico y tratamiento en el ámbito de la salud. Estos datos necesitan ser distribuidos y compartidos frecuentemente entre diferentes actores como proveedores de salud, compañías de seguros, farmacias, investigadores, familias de los pacientes, etc. Este hecho supone un gran desafío a la hora de mantener actualizado y/o disponible el historial médico del paciente. Almacenar y compartir datos entre múltiples entidades, así como mantener el control de acceso a través de numerosos consentimientos sólo complica el proceso del tratamiento de un paciente. Por lo tanto, tener acceso al historial completo puede ser crucial para el correcto tratamiento de un paciente: por ejemplo, en el caso de un paciente con cáncer, es fundamental conocer con exactitud las dosis de radiación suministradas o los resultados de los análisis de laboratorio para continuar con el tratamiento. [59]

Un problema clave en los sistemas sanitarios actuales es la falta de enlaces seguros que puedan conectar todos los sistemas de salud independientes entre sí para establecer una red accesible de extremo a extremo a la vez que se protege la privacidad de los profesionales sanitarios y de los propios pacientes. Aunque ya existen unos estándares de datos que proporcionan una interoperabilidad básica para el intercambio de datos entre diferentes sistemas, este nivel de interoperabilidad se limita a los estándares de la implementación y requiere el mapeo de datos entre sistemas en prácticamente todos los casos. El mantenimiento de dichos sistemas también es una tarea difícil de lograr ya que un cambio de interfaz en un sistema requiere a su vez un cambio en la red de confianza para adaptarse a dicho cambios.

Otro de los problemas fundamentales en el intercambio de información sanitaria es la identificación del paciente. Es decir, poder encontrar a un paciente en una base de datos sanitaria utilizando un conjunto único de datos. A pesar del creciente esfuerzo de desarrollo de bases de datos únicas, aún existen problemas relativos a la duplicación de los registros, así como la existencia de datos médicos incompletos o incorrectos. [60]

En consecuencia, es necesario idear un sistema de intercambio de información médica que resuelva los problemas descritos anteriormente. Y es por ello que la posibilidad de utilizar Blockchain para la gestión de datos sanitarios ha suscitado mucho interés tanto en la industria como en el mundo académico, ya que esta tecnología permitiría garantizar la seguridad y la privacidad de los datos, así como su disponibilidad respecto a la política de control de acceso definida por el paciente. [59]

Por lo tanto, en este trabajo se propone la adaptación del sistema de identidad soberana desarrollado anteriormente con Hyperledger Indy para el ámbito de la medicina. Esto es debido a que, tal y como se ha comentado en el apartado 6.1.1, es necesario emplear una Blockchain permisionada ya que, una cadena de bloques pública no asegura la confianza en un sistema, por lo que sería necesario implementar un algoritmo de consenso muy costoso. Además, las Blockchain permisionadas son mucho más escalables que las cadenas públicas y su crecimiento puede ser controlado.

Este sistema aseguraría los siguientes tres objetivos clave:

1. Garantizar el almacenamiento y traspaso seguro de los datos respetando la actual normativa de protección de datos de la UE.
2. Asegurar la unicidad y privacidad de la identidad del paciente.
3. Otorgar al paciente control y derecho de decisión sobre sus datos.

Finalmente, cabe recordar que este sistema consiste en una implementación del concepto de identidad soberana, por lo que el paciente posee un control absoluto sobre sus datos médicos, siendo este un intermediario a la hora de realizarse un intercambio de información médica ente dos instituciones. Es decir, no se trata de un mero sistema de intercambio basado en Blockchain entre dos o más organizaciones, ajeno al paciente.

6.2 Descripción y caso de uso del sistema

El sistema que se va a poner en marcha contará con tres actores:

- **Denis:** un paciente.
- **Osakidetza - Servicio Vasco de Salud:** sanidad pública del País Vasco.
- **Assistance publique – Hôpitaux de Paris (AP-HP):** sistema de hospitales de París (Francia).

El caso de uso a implementar es el siguiente:

- El paciente Denis, ciudadano de País Vasco - España, tiene alergia a un antibiótico muy usado en el tratamiento de las infecciones respiratorias llamado "Claritromicina". Dicho paciente estaba pasando sus vacaciones en París - Francia, cuando contrajo una infección de garganta. Para evitar que los médicos franceses le den una receta médica que contenga el antibiótico mencionado anteriormente, Denis usará la novedosa aplicación SSI de MED-ID para compartir la información relativa a su alergia. Para ello, tendrá que recoger su historial médico de Osakidetza, a través de la misma aplicación.

Por lo tanto, el diagrama actualizado del sistema quedaría de la siguiente manera:

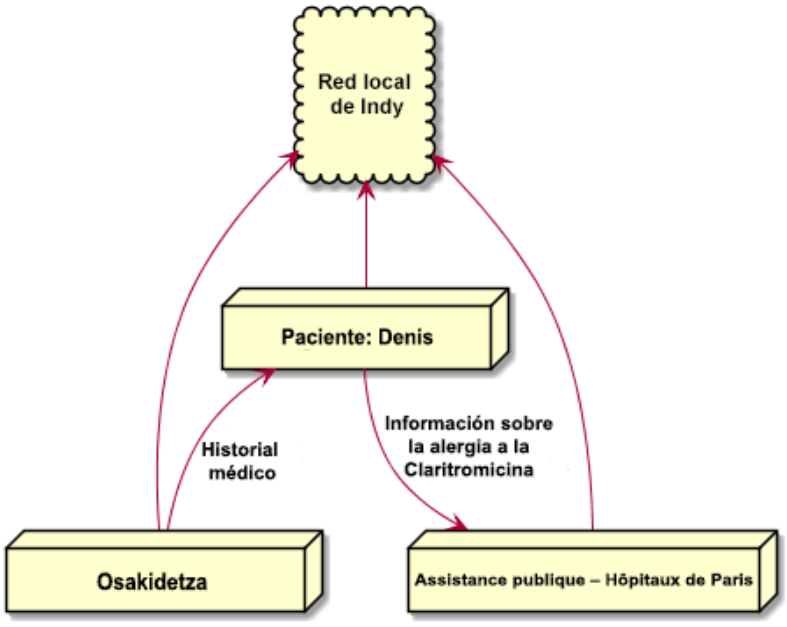

Figura 42 - Diagrama actualizado del sistema

En cuanto al expediente médico del paciente (Denis), este estará basado en el siguiente documento:

FICHA CLINICA Y ESTETICA

Motivo de Consulta: .....

Diagnóstico: .....

Tratamiento: .....

Sesiones: .....Nº sesión: .....

ANTECEDENTES PERSONALES:

Nombre: .....Nº ficha:.....

Dirección: .....

Telefono de contacto: .....email:.....

Fecha de nacimiento: .....Edad.....

Ocupación u oficio: .....

Nº de hijos: .....Tipo parto: .....

ANTECEDENTES CLINICOS:

Enfermedades: .....

Medicamentos que toma actualmente: .....

Antecedentes enfermedades crónicas: .....

Antecedentes quirúrgicos: .....

Uso de Implantes o dispositivos.....Tipo: .....

Alergias: .....

Figura 43 - Plantilla de expediente médico [61]

6.3 Puesta en marcha

La puesta en marcha del sistema se realiza de la misma manera que en el apartado 6.2, con los comandos:

$ sudo ./manage build

$ sudo ./manage up

Ahora, se puede acceder a cada uno de los agentes desde el navegador web, en este caso, Mozilla Firefox.

Los puertos donde se ejecuta cada agente son:

- **Denis:** puerto 3000
- **Osakidetza - Servicio Vasco de Salud:** puerto 3002
- **Assistance publique – Hôpitaux de Paris (AP-HP):** puerto 3003

Al entrar en cualquiera de los puertos expuestos anteriormente, se muestra una pantalla de logueo:

Figura 44 - Login de MED-ID

Tal y como se ha definido en el fichero ***docker-compose.yaml***, las credenciales de acceso para cada agente son:

- **Denis:**
  - Usuario: denis
  - Contraseña: 123
- **Osakidetza - Servicio Vasco de Salud:**
  - Usuario: osakidetza
  - Contraseña: 123

- **Assistance publique – Hôpitaux de Paris (AP-HP):**
    - Usuario: aphp
    - Contraseña: 123

En un principio, Denis, al ser residente del País Vasco, ya tiene una relación establecida con Osakidetza, así como su historial médico (historial que Osakidetza ha generado desde el apartado "Emisión").

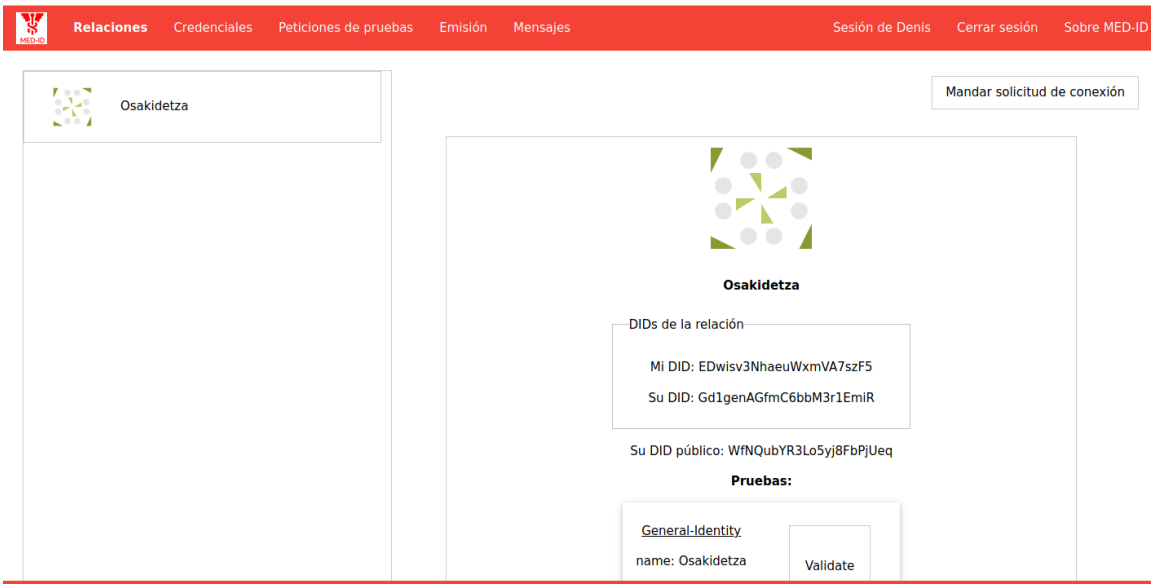

Figura 45 - Relación entre Denis y Osakidetza

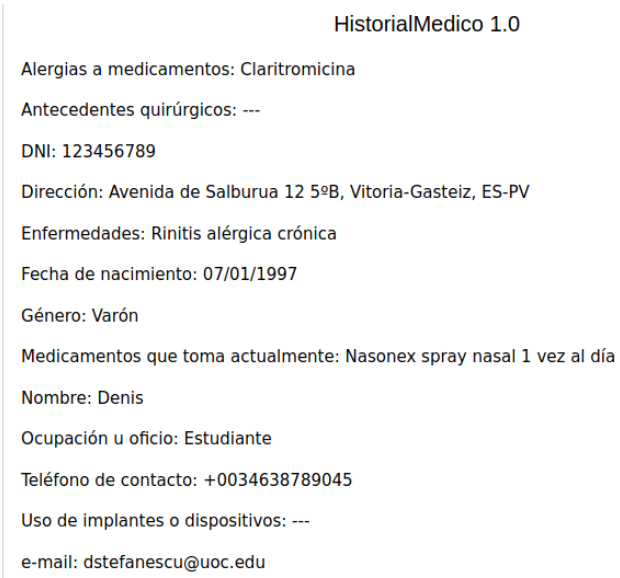

Figura 46 - Historial médico de Denis

Ahora, debido a que Denis se encuentra en París, Francia, la sanidad parisina necesita acceder a su historial médico para comprobar sus datos, en concreto el dato de las alergias a los medicamentos, ya que Denis es alérgico a un tipo de antibiótico muy utilizado a la hora de tratar las infecciones respiratorias como la que Denis contrajo en París. En un sistema de identidad soberana el usuario puede elegir qué datos compartir, pero en este caso sería preciso no omitir ningún dato y compartir todo el historial médico con la sanidad parisina.

Para ello, Assistance publique – Hôpitaux de Paris (AP-HP) necesita mandarle a Denis una petición de pruebas (proof request) con los atributos del historial médico.

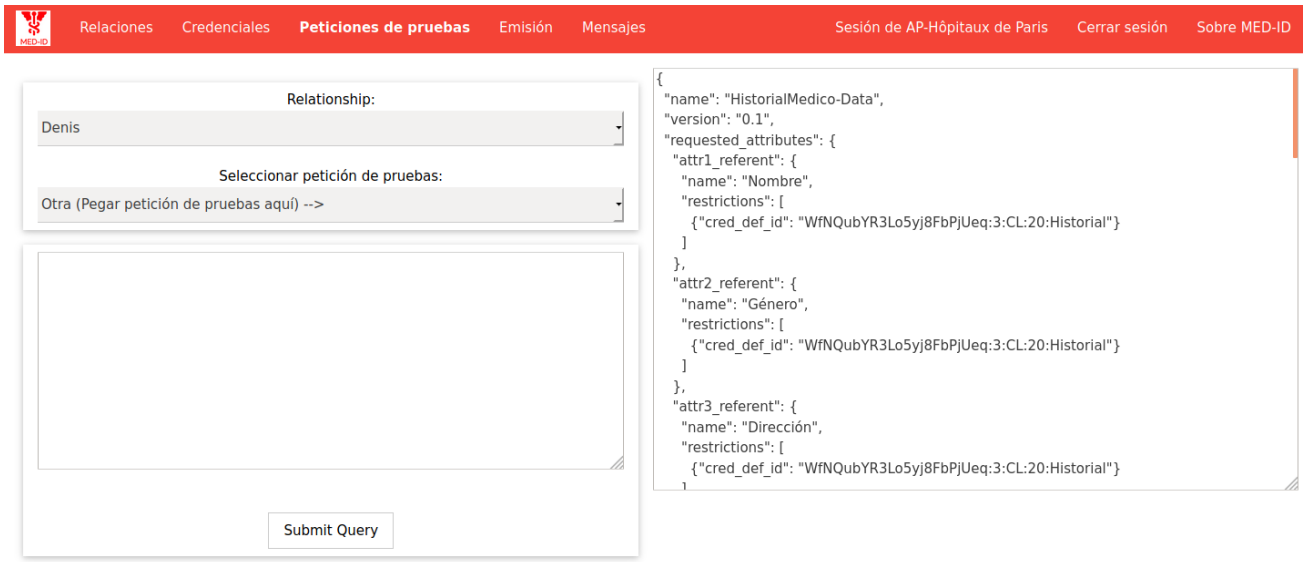

Figura 47 - Petición de pruebas - expediente médico

Denis recibirá una petición de acceso a su expediente médico en su aplicación:

Figura 48 - Petición de acceso al expediente médico

Ahora, en el apartado de Relaciones de Assistance publique – Hôpitaux de Paris (AP-HP), en la relación con el paciente Denis, aparecerá su expediente médico, y este podrá ser validado:

Figura 49 - Acceso de la sanidad parisina al expediente de Denis

Ahora la sanidad parisina puede conocer todo su historial, incluida la alergia a la claritromicina:

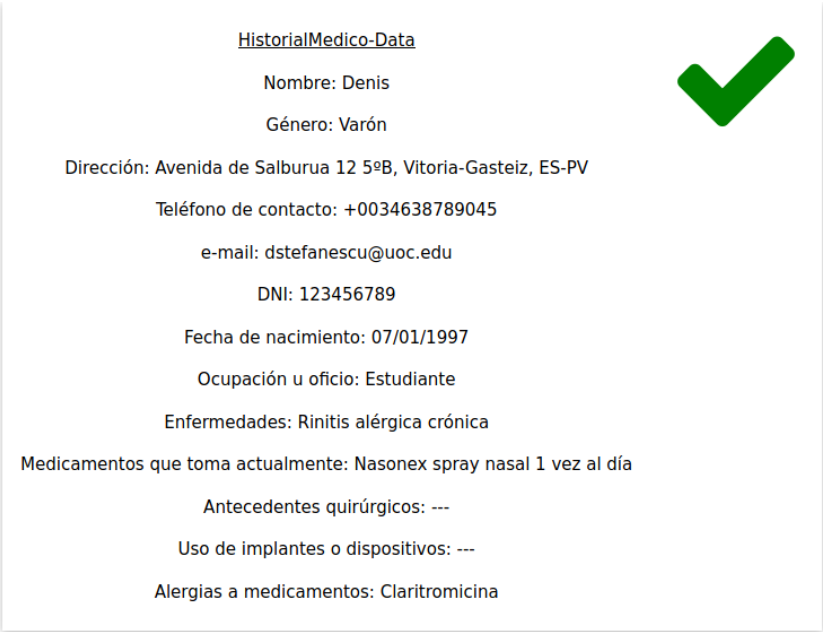

Figura 50 - Historial de Denis validado por la sanidad parisina

Tal y como se ha explicado anteriormente en el apartado de "Introducción", cabe recordar que este sistema consiste en una implementación del concepto de identidad soberana, por lo que el paciente posee un control absoluto sobre sus datos médicos, **siendo este un intermediario a la hora de realizarse un intercambio de información médica ente dos instituciones**. Es decir, no se trata de un mero sistema de intercambio basado en Blockchain entre dos o más organizaciones, ajeno al paciente.

Para parar el sistema, hay que introducir el comando *$ sudo ./manage down*.

```
denis@ubuntu:~/TFM_PROYECTO2$ sudo ./manage down
[sudo] password for denis:
WARNING: The RUST_LOG variable is not set. Defaulting to a blank string.
WARNING: The TEST_POOL_IP variable is not set. Defaulting to a blank string.
Stopping tfm_proyecto2_AP-HP_1      ... done
Stopping tfm_proyecto2_osakidetza_1 ... done
Stopping tfm_proyecto2_denis_1      ... done
Stopping tfm_proyecto2_webserver_1  ... done
Stopping tfm_proyecto2_node1_1      ... done
Stopping tfm_proyecto2_node4_1      ... done
Stopping tfm_proyecto2_node2_1      ... done
Stopping tfm_proyecto2_node3_1      ... done
Removing tfm_proyecto2_AP-HP_1      ... done
Removing tfm_proyecto2_osakidetza_1 ... done
```

Figura 51 - Parar el sistema de MED-ID

6.4 Evaluación de la identidad digital en MED-ID

Este sistema facilita la conexión entre distintos sistemas informáticos sin la necesidad de tener que desarrollar una macro-base de datos única y centralizada o sistemas complejos de intercambio de información entre sistemas radicalmente distintos que además puedan ser fácilmente manipulables o cuya información pueda ser robada fácilmente por piratas informáticos.

Además, el sistema MED-ID le asegura al paciente un control total sobre su información, pudiendo este compartir una parte o la totalidad de sus datos médicos con la institución que desee, además de tener la posibilidad de revocar en cualquier momento cualquier intercambio de datos. Los aspectos comentados en este párrafo hacen que el sistema MED-ID se adecúe perfectamente al actual Reglamento Europeo de Protección de Datos (GDPR), ya que dicho Reglamento establece que los usuarios deben tener un control total sobre sus datos.

Finalmente, otro aspecto importante a evaluar es el rendimiento del sistema. Un aspecto muy positivo en cuanto al rendimiento es el empleo de una Blockchain permisionada, y por consiguiente la no-necesidad de implementar un algoritmo de consenso costoso como la prueba de trabajo (PoW). Además, según un estudio similar sobre el empleo de Blockchain en la transmisión de datos en el sector de la salud [62], al contar con el método del árbol de Merkle para el agrupamiento de los datos (condición que cumple Hyperledger Indy), se logra una complejidad de cálculo $\boldsymbol{O}(log_2\, n)$, lo que supone una gran ventaja de rendimiento.

8. Informe de recursos destinados al proyecto

6.5 Introducción

En este punto se analizarán hasta qué grado los plazos temporales y recursos planificados al inicio de la aplicación difieren de los plazos y recursos reales que se han destinado al mismo.

En los siguientes apartados se mostrarán los tiempos reales del desarrollo y un análisis de la diferencia de tiempos y el porqué de los mismos.

6.6 Dedicación temporal real al proyecto

Tabla 4 - Temporización real del proyecto

Iteración	Actividad	Horas
1	Realizar la planificación del proyecto	4
2	Introducción de conceptos básicos previos: Blockchain, identidad digital e identidad soberana	15
3	Análisis de las soluciones existentes para la gestión de la identidad en Blockchain	50
4	Diseño y desarrollo de un método o aplicación que asegure la identidad o proporcione garantía sobre ella (sistema de identidad soberana)	70
5	Pruebas realizadas sobre el sistema desarrollado	8
6	Análisis de la identidad soberana en el ámbito concreto de la medicina	10
7	Puesta en marcha de un DLT para el intercambio de expedientes médicos	40
8	Documentación y presentación del proyecto	30

TOTAL HORAS INVERTIDAS: 227

Este proyecto tiene un valor de 9 créditos ECTS, siendo 1 crédito equivalente a 25 horas de trabajo. Es decir, 9 créditos requieren mínimo 225 horas de dedicación por parte del alumno. Este objetivo se ha cumplido satisfactoriamente, ya que la suma total de horas invertidas son 227 horas, 3 horas por encima del mínimo exigido.

6.7 Comparativa entre tiempos reales y estimados

Tabla 5 - Comparativa entre tiempo estimado y tiempo invertido

Iteración	Actividad	Horas invertidas	Horas estimadas	Diferencia (invertidas – estimadas)
1	Realizar la planificación del proyecto	4	2	+ 2
2	Introducción de conceptos básicos previos: Blockchain, identidad digital e identidad soberana	15	15	0
3	Análisis de las soluciones existentes para la gestión de la identidad en Blockchain	50	40	+ 10
4	Diseño y desarrollo de un método o aplicación que asegure la identidad o proporcione garantía sobre ella (sistema de identidad soberana)	70	65	+ 5
5	Pruebas realizadas sobre el sistema desarrollado	8	15	- 7
6	Análisis de la identidad soberana en el ámbito concreto de la medicina	10	15	- 5
7	Puesta en marcha de un DLT para el intercambio de expedientes médicos	40	50	- 10
8	Documentación y presentación del proyecto	30	20	+ 10

Como se puede observar en la tabla anterior, hay algunas tareas que no han requerido tantas horas como las que se habían estimado en un principio (diferencia negativa). La tarea que menos horas ha requerido comparado con lo que se había estimado es la puesta en marcha de un DLT para el intercambio de expedientes médicos. Esto se debe a que ya se partía de la base del sistema desarrollado en el capítulo 6 (UOC-ID), por lo que lo único que se requería era realizar una adaptación y mejora del mismo -> MED-ID.

En cuanto a las tareas que han requerido más dedicación que la que se había previsto, las mayores diferencias de tiempo están en las 3 tareas siguientes:

- El análisis de las soluciones existentes para la gestión de la identidad en Blockchain ha requerido 10 horas más de lo previsto debido a la gran cantidad de información disponible y su complejidad.
- La tarea de la documentación también ha requerido aproximadamente 10 horas más de lo estimado.
- Por último, la tarea de diseñar y desarrollar de un método o aplicación que asegure la identidad o proporcione garantía sobre ella (sistema de identidad soberana) también ha requerido más tiempo de lo previsto (5 horas), sobre todo gracias a las dificultades encontradas en la programación (errores, funcionamiento anormal…).

9. Conclusiones

En este trabajo se han analizado múltiples soluciones para la gestión de la identidad de manera descentralizada y soberana, y, además, se han puesto en marcha dos sistemas de identidad soberana para dos casos de uso radicalmente distintos, lo que sirvió para demostrar el potencial de la tecnología Blockchain en el ámbito de la identidad digital. Los sistemas analizados han demostrado sin duda alguna ser aptos para la construcción de aplicaciones innovadoras para gestionar la identidad del futuro. Además, dichas soluciones cuentan con un importante respaldo por parte de la comunidad científica y de las grandes empresas del sector de la tecnología, por lo que el ámbito de la gestión descentralizada de la identidad tiene un futuro muy prometedor. El concepto de identidad soberana no solo cambiaría para bien Internet, otorgándole al usuario más privacidad y control sobre sus datos personales, sino que podría servir para reducir la complejidad y los costes de los actuales sistemas, sin olvidar también la importante mejora de la seguridad de los datos, seguridad que actualmente es muy débil.

El objetivo principal de este proyecto ha sido estudiar y evaluar la identidad digital soberana en Blockchain (DLT) en un entorno como es el intercambio de expedientes médicos entre distintos países europeos (teniendo en cuenta GDPR), además de poner en marcha un sistema de identidad soberana para dicho intercambio. Para ello se ha adaptado el sistema desarrollado con anterioridad (capítulo 6) -> UOC-ID, convirtiéndolo en MED-ID.

Después de poner en marcha y evaluar el sistema MED-ID, se ha observado como un sistema de identidad soberana para el intercambio de información médica resolvería varios problemas muy comunes de los sistemas actuales:

- El sistema puesto en marcha facilitaría la conexión entre distintos sistemas informáticos sin la necesidad de tener que desarrollar una macro-base de datos única y centralizada o sistemas complejos de intercambio de información entre sistemas radicalmente distintos que además puedan ser fácilmente manipulables o cuya información pueda ser robada fácilmente por piratas informáticos.
- El sistema MED-ID le asegura al paciente un control total sobre su información, pudiendo este compartir una parte o la totalidad de sus datos médicos con la institución que desee, además de tener la posibilidad de revocar en cualquier momento cualquier

intercambio de datos. Los aspectos comentados en este párrafo hacen que el sistema MED-ID se adecúe perfectamente al actual Reglamento Europeo de Protección de Datos (GDPR), ya que dicho Reglamento establece que los usuarios deben tener un control total sobre sus datos.

- El rendimiento del sistema también es positivo gracias al empleo de una Blockchain permisionada, y por consiguiente la no-necesidad de implementar un algoritmo de consenso costoso como la prueba de trabajo (PoW). Además, al contar con el método del árbol de Merkle para el agrupamiento de los datos, se logra una complejidad de cálculo ***O****(log2 n)*, lo que supone una gran ventaja de rendimiento.

Sin embargo, la tecnología Blockchain y la gestión de la identidad soberana aún no tienen el grado de madurez suficiente para ser utilizadas a gran escala. Hay muchos aspectos de diseño y funcionamiento de estos sistemas que aún no están del todo claros. Además, todas las soluciones analizadas aún están en fase de continuo desarrollo y mejora. La puesta en marcha de este tipo de soluciones en entornos reales y críticos requiere más inversión por parte de las instituciones tanto públicas como privadas e investigación por parte de la comunidad científica. Es fundamental acelerar este proceso especialmente en ámbitos como el intercambio de información médica, ya que disponer de un sistema de intercambio rápido, sencillo y seguro podría contribuir a salvar vidas.

6.8 Trabajo futuro

El trabajo futuro se puede dividir en dos partes: la parte teórica y la parte práctica.

En lo que respecta la parte teórica, esta se puede extender enormemente, ya que el concepto de identidad soberana es relativamente nuevo y sigue creciendo cada día gracias a los investigadores en este ámbito y a los avances de la tecnología, además de la creciente preocupación por parte de los individuos u organizaciones acerca de la privacidad y la gestión de los datos y de la identidad en internet. En cuanto al ámbito del intercambio de expedientes médicos mediante DLTs, resulta de vital importancia seguir investigando y desarrollando mecanismos que ayuden a la gestión de la identidad digital (identidad soberana), ya que es

necesario profundizar en la colaboración entre regiones y/o países para garantizar un mayor bienestar de la ciudadanía.

Los sistemas que se han puesto en marcha, UOC-ID y MED-ID requieren algunas mejoras importantes como, por ejemplo:

- La mejora de la interfaz gráfica, especialmente de los apartados de peticiones de pruebas y emisión de credenciales. Actualmente el mecanismo es muy rudimentario, ya que se basa en copiar y pegar textos en formato JSON. Se podría mejorar la interfaz implementando un generador de credenciales y de *proof requests* más sencillo y visual.
- El almacenamiento de los datos se define de forma manual. En un escenario real, los datos necesitarían estar almacenados de forma segura en los sistemas de cada actor. No es recomendable el almacenamiento de credenciales dentro de la cadena de bloques por motivos de privacidad y seguridad.
- La red es local y se está simulando con Docker. Hace falta poner en marcha una red y una cadena de bloques real, para poder realizar una evaluación más completa.

Acrónimos

DLT - Distributed ledger technology

GDPR - General Data Protection Regulation (UE)

SSI - Self-Sovereign Identity

PEC - Prueba de evaluación continuada

P2P - Peer to peer

MIT - Instituto de Tecnología de Massachussets

BD - Base de datos

ID - Identificador

UE - Unión Europea

ZKP - Prueba de Conocimiento Cero

UOC - Universitat Oberta de Catalunya

TIC - Tecnologías de la Información y de las Comunicaciones

DID - Identificadores descentralizados

JSON - JavaScript Object Notation

URI - Identificador de recursos uniforme

URL - Localizador de recursos uniforme

JWT - JSON Web Token

OAuth - Open Authorization

RSK - Rootstock

DPKI - Infraestructura de calve pública descentralizada

GW - Gateway

IPFS - Sistema de archivos interplanetarios (P2P)

DGT - Dirección General de Tráfico (España)

VC - Credenciales verificables

Referencias

[1] “What is Blockchain? The complete guide”, ComputerWorld, enero 2019. Enlace: https://www.computerworld.com/article/3191077/what-is-blockchain-the-complete-guide.html

[2] “Blockchain”, Wikipedia en inglés. Enlace: https://en.wikipedia.org/wiki/Blockchain

[3] “El concepto de prueba de trabajo”, ElBitcoin.org, agosto 2014. Enlace: https://elbitcoin.org/el-concepto-de-prueba-de-trabajo/

[4] “Blockchain: estudio de alternativas e implementaciones”, Denis Ionut Stefanescu (UPV/EHU), junio 2018. Enlace: https://lsi.vc.ehu.eus/pablogn/docencia/PFC/Memoria%20TFG%20-%20Denis%20Ionut%20Stefanescu.pdf

[5] “Hyperledger”, AprendeBlockchain. Enlace: https://aprendeBlockchain.wordpress.com/hyperledger/

[6] “Hyperledger”, AprendeBlockchain. Enlace: https://aprendeBlockchain.wordpress.com/hyperledger/

[7] “Blockchain Híbrida - lo mejor de ambos mundos”, 101blockchains, diciembre 2018. Enlace: https://101blockchains.com/es/blockchain-hibrida/

[8] “¿Qué es la identidad digital?”, Gobierno de Canarias. Enlace: http://www3.gobiernodecanarias.org/medusa/ecoescuela/seguridad/identidad-digital-profesorado/que-es-la-identidad-digital/

[9] “Identidad digital, reputación online, el derecho al olvido y cómo borrar contenido de Internet”, Un Community Manager, enero 2018. Enlace: https://www.uncommunitymanager.es/identidad-digital/

[10] “¿Qué es la identidad soberana?”, Bit2me Academy. Enlace https://academy.bit2me.com/que-es-la-identidad-soberana/

[11] Stokkink, Q., & Pouwelse, J. (2018, July). Deployment of a blockchain-based self-sovereign identity. In 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) (pp. 1336-1342). IEEE.

[12] "Cómo Blockchain puede devolver el control de identidad digital a los usuarios y el modelo de identidad soberana", Aprende Blockchain. Enlace: https://aprendeblockchain.wordpress.com/como-blockchain-puede-devolver-el-control-de-identidad-digital-a-los-individuos-y-el-modelo-de-identidad-soberana

[13] Abraham, A. (2017). Self-Sovereign Identity.

[14] "¿Qué es la identidad digital soberana?", Open Trends, marzo 2019. Enlace: https://www.opentrends.net/es/articulo/que-es-la-identidad-digital-soberana

[15] "Una identidad digital fuerte es la clave para la protección de los usuarios", El Mundo, octubre 2019. Enlace: https://www.elmundo.es/cataluna/2019/10/29/5db82e9c21efa0634c8b45cb.html

[16] "1.2 Blockchain: pros y contras", Medium, mayo 2018. Enlace: https://medium.com/@alvarosb999/1-2-blockchain-pros-y-contras-a0ec53f786fb

[17] "What is the Difference between Blockchain and Database?", Atoz Markets, marzo 2018, Enlace: https://atozmarkets.com/news/difference-between-blockchain-and-database/

[18] "Reglamento General de Protección de Datos", Wikipedia en español. Enlace: https://es.wikipedia.org/wiki/Reglamento_General_de_Protecci%C3%B3n_de_Datos

[19] "Prueba de conocimiento cero", Wikipedia en español. Enlace: https://es.wikipedia.org/wiki/Prueba_de_conocimiento_cero

[20] "Grandes compañías españolas constituyen el consorcio Alastria para desarrollar el ecosistema 'Blockchain' en España", BBVA, octubre 2017. Enlace: https://www.bbva.com/es/grandes-companias-espanolas-constituyen-consorcio-alastria-desarrollar-ecosistema-blockchain-espana/

[21] “Alastria – Spain's National Token & Identity Blockchain Consortium?”, Blockchain Exchange Guide, octubre 2017. Enlace: https://bitcoinexchangeguide.com/alastria/

[22] “Sobre Alastria”, Alastria. Enlace: https://alastria.io/sobre_alastria

[23] Página de inicio de Alastria, Alastria. Enlace: https://alastria.io/home

[24] “Blockchain - La tecnología que transforma tu negocio”, Tecnalia. Enlace: https://www.spri.eus/euskadinnova/es/enpresa-digitala/agenda/arabatic-2019-semana-digital-alava/documentos/3319.aspx

[25] “Presentación de Alastria”, Alastria. Enlace: https://alastria.io/assets/docs/Alastria_Presentacio%CC%81n_general_.pdf

[26] “ID Alastria”, Alastria. Enlace: https://alastria.io/id-alastria/

[27] “Así avanza el proyecto Alastria ID”, Medium, mayo 2018. Enlace: https://medium.com/@alastria_es/as%C3%AD-avanza-el-proyecto-alastria-id-c206aa649770

[28] “El trabajo de identidad digital en Alastria”, Tribalyte, agosto 2019. Enlace: https://tech.tribalyte.eu/blog-identidad-digital-alastria

[29] “Alastria: presentación corporativa”, Alastria, abril 2019. Enlace: https://alastria.io/wp-content/uploads/2019/04/2019-04-23_Alastria-Presentaci%C3%B3n-corporativa_v00.08.pdf

[30] “Sovrin™: A Protocol and Token for Self-Sovereign Identity and Decentralized Trust”, Sovrin, enero 2018. Enlace: https://sovrin.org/wp-content/uploads/Sovrin-Protocol-and-Token-White-Paper.pdf

[31] “Sovrin Blockchain Digital Identity Platform Review”, Cryptotomorrow, septiembre 2018. Enlace: https://www.cryptomorrow.com/2018/09/07/sovrin-blockchain-digital-identity-platform-review/

[32] “Sovrin Review: A Protocol And Token For Self-Sovereign Identity And Decentralized Trust”, Medium, diciembre 2018. Enlace: https://medium.com/@EVALUAPE1/sovrin-review-a-protocol-and-token-for-self-sovereign-identity-and-decentralized-trust-97c524f8bde

[33] “Sovrin: What Goes on the Ledger?”, Sovrin Foundation & Evernym, abril 2017. Enlace: https://sovrin.org/wp-content/uploads/2018/10/What-Goes-On-The-Ledger.pdf

[34] Dunphy, P., & Petitcolas, F. A. (2018). A first look at identity management schemes on the Blockchain. IEEE Security & Privacy, 16(4), 20-29.

[35] “Sovrin Provisional Trust Framework”, Sovrin Foundation, marzo 2017. Enlace: https://sovrin.org/wp-content/uploads/2017/06/SovrinProvisionalTrustFramework2017-03-22.pdf

[36] “Sovrin review”, Cryptocalibur. Enlace: https://cryptocalibur.com/sovrin-ico-review/

[37] “How does Sovrin work?”, Sovrin, diciembre 2018. Enlace: https://sovrin.org/faq/how-does-sovrin-work-2/

[38] “Hyperledger: la Blockchain privada que todos tenemos que conocer”, El Economista, enero 2018. Enlace: https://www.eleconomista.es/economia/noticias/8899454/01/18/Hyperledger-la-Blockchain-privada-que-todos-tenemos-que-conocer.html

[39] “Hyperledger members”, Hyperledger. Enlace: https://www.hyperledger.org/members

[40] Página principal de Hyperledger. Enlace: https://www.hyperledger.org/

[41] “What Is Hyperledger Indy?”, Sovrin, diciembre 2018. Enlace: https://sovrin.org/faq/what-is-hyperledger-indy/

[42] Hyperledger Indy Docs. Enlace: https://indy.readthedocs.io/en/latest/

[43] “uPort”, LDAP Wiki, octubre 2018. Enlace: https://ldapwiki.com/wiki/UPort

[44] “All you need to know about uPort Identity management”, Medium, febrero 2019. Enlace: https://medium.com/@hamzamaslah/all-you-need-to-know-about-uport-identity-management-3fc49db25332

[45] uport-transports, Github. Enlace: https://github.com/uport-project/uport-transports

[46] uport-credentials, Github. Enlace: https://github.com/uport-project/uport-credentials

[47] "El 'Gas' en Ethereum", Mi Ethereum. Enlace: https://www.miethereum.com/ether/gas/

[48] Imágenes sobre uPort, All Data Sheet. Enlace. https://www.alldatasheet.net/view_image.jsp?components=UPORT

[49] "Polling (informática)", Wikipedia en español. Enlace: https://es.wikipedia.org/wiki/Polling_(red_inform%C3%A1tica)

[50] "Qué son las DApps y por qué serán cada vez más importantes", BBVA, septiembre 2018. Enlace: https://www.bbva.com/es/que-son-las-dapps-y-por-que-seran-cada-vez-mas-importantes/

[51] "Working together for better self-sovereign identity: Civic, SelfKey, and Peer Mountain", Medium, febrero 2018. Enlace: https://medium.com/peermountain/working-together-for-better-self-sovereign-identity-civic-selfkey-and-peer-mountain-282bca9a8e4a

[52] "Technical Review of Civic's Secure Identity Platform", Scott Brady, febrero 2018. Enlace: https://www.scottbrady91.com/Blockchain-Identity/Technical-Review-of-Civics-Secure-Identity-Platform

[53] "Secure Identity Platform", Civic. Enlace: https://www.civic.com/products/secure-identity-platform/

[54] "OAuth", Wikipedia en español. Enlace: https://es.wikipedia.org/wiki/OAuth

[55] "Civic Whitepaper", Civic. Enlace: https://tokensale.civic.com/CivicTokenSaleWhitePaper.pdf

[56] "RSK", Wikipedia en español. Enlace: https://es.wikipedia.org/wiki/RSK

[57] "Acciones Alastria Id con Gateway", taiga.io, marzo 2018. Enlace: https://tree.taiga.io/project/marcossanlab-alastria-identity-core-team/issue/1

[58] Hileman, G., & Rauchs, M. (2017). Global blockchain benchmarking study. Cambridge Centre for Alternative Finance, University of Cambridge, 122.

[59] Dubovitskaya, A., Xu, Z., Ryu, S., Schumacher, M., & Wang, F. (2017). Secure and trustable electronic medical records sharing using blockchain. In AMIA Annual Symposium Proceedings (Vol. 2017, p. 650). American Medical Informatics Association.

[60] Zhang, P., Schmidt, D. C., White, J., & Lenz, G. (2018). Blockchain technology use cases in healthcare. In Advances in Computers (Vol. 111, pp. 1-41). Elsevier.

[61] Plantilla expediente médico. Enlace: https://i.pinimg.com/originals/68/02/74/6802746ad715c1692ecdf8df5861d1e6.jpg

[62] Liang, X., Zhao, J., Shetty, S., Liu, J., & Li, D. (2017, October). Integrating blockchain for data sharing and collaboration in mobile healthcare applications. In 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC) (pp. 1-5). IEEE.

ANEXO I: Guía de instalación

Instalación de Ubuntu 18.04 LTS

Al arrancar la máquina virtual con la imagen de Ubuntu puesta aparecerá la siguiente pantalla. En este caso de va a optar por la opción de la derecha, la instalación de Ubuntu en el disco duro virtual.

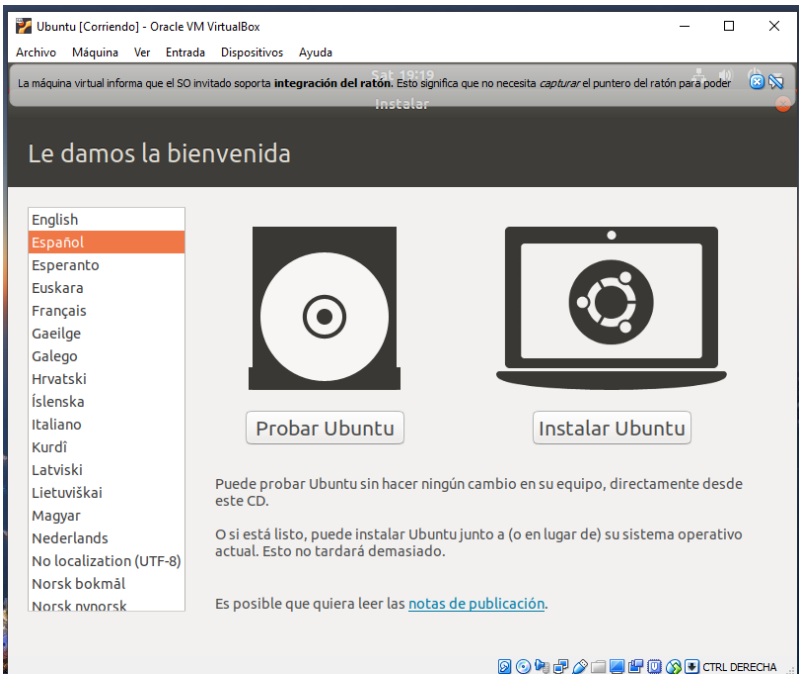

Figura 52 - Instalación de Ubuntu (1)

Posteriormente se tiene que elegir el idioma del sistema y del teclado. En este caso, español.

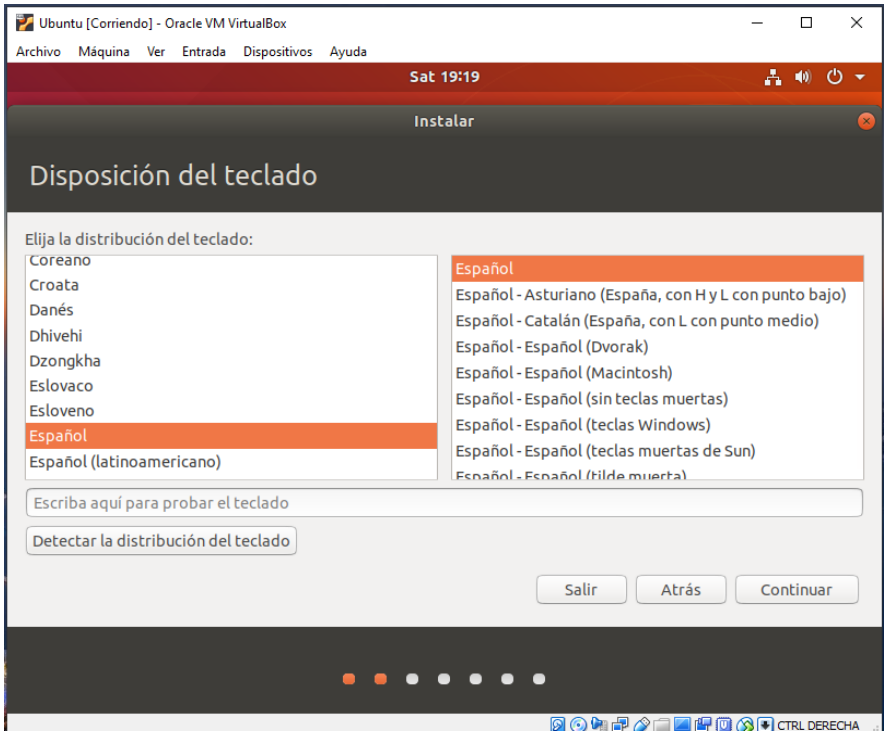

Figura 53 - Instalación de Ubuntu (2)

Para evitar problemas se va a optar por una instalación completa del SO.

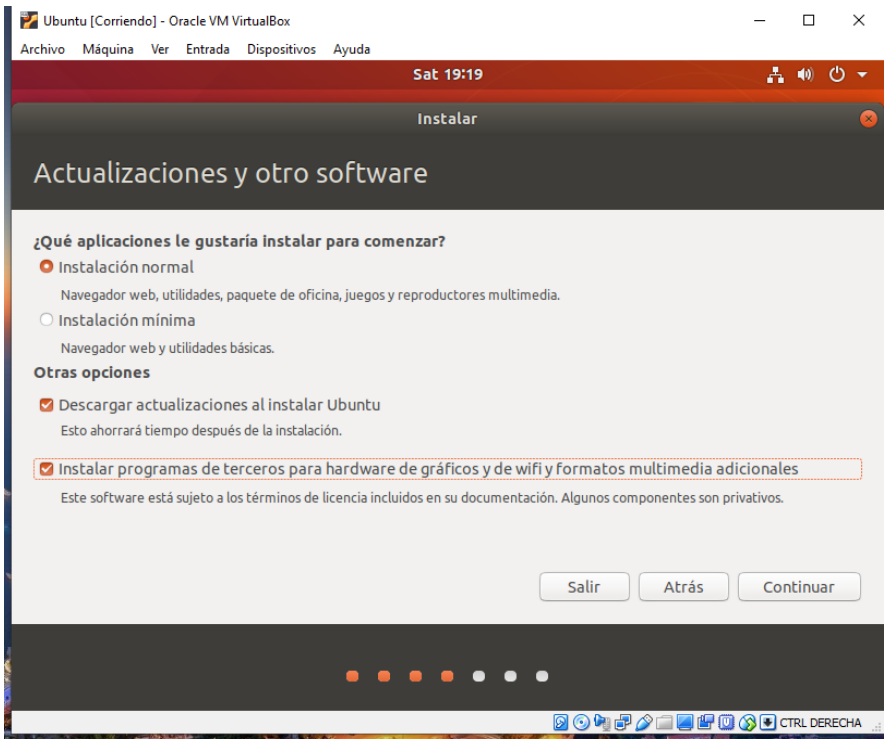

Figura 54 - Instalación de Ubuntu (3)

Al tratarse de una instalación dentro de una máquina virtual, se va a usar todo el disco duro que se le ha asignado a dicha MV.

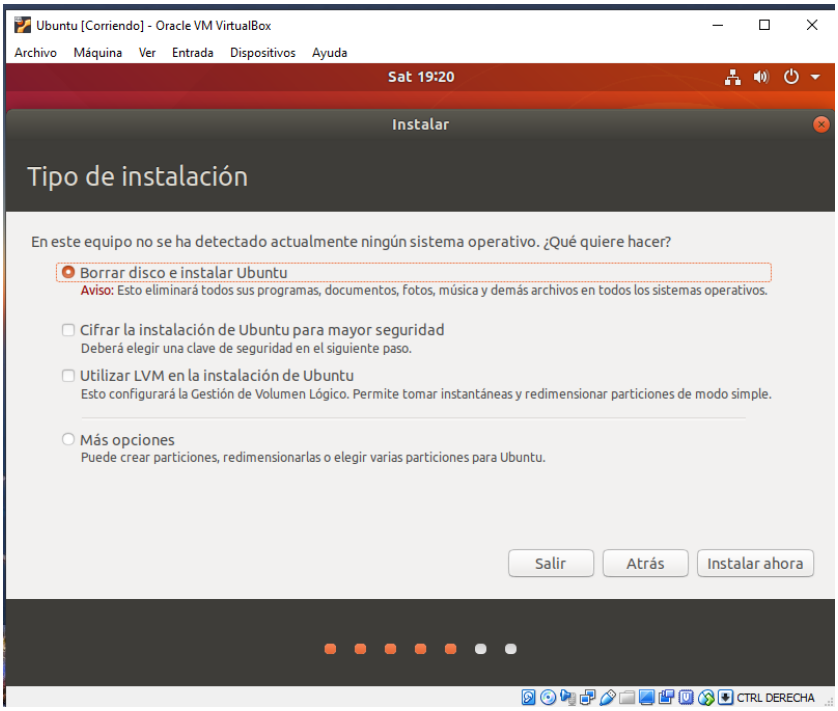

Figura 55 - Instalación de Ubuntu (4)

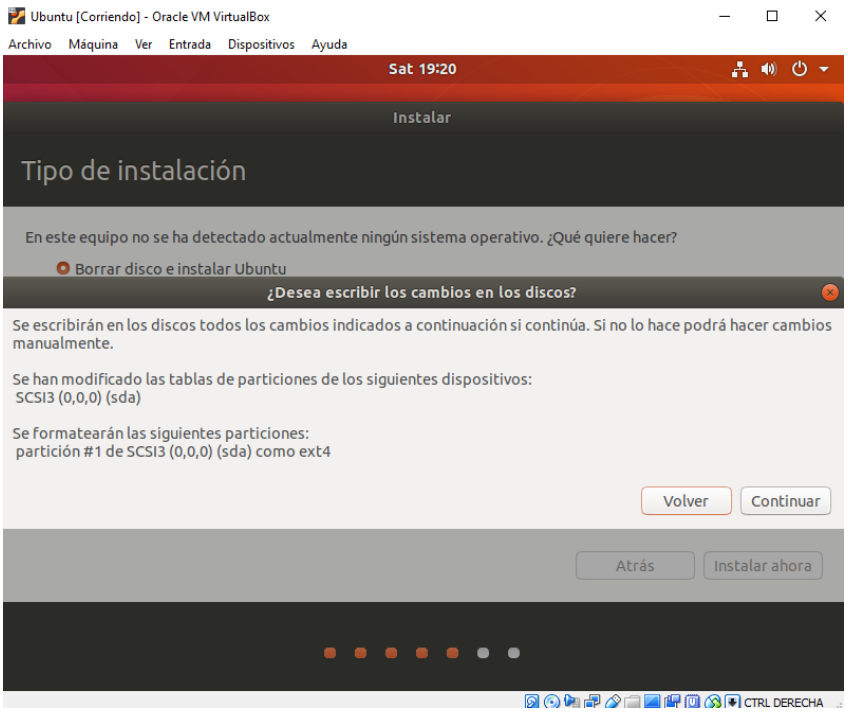

Figura 56 - Instalación de Ubuntu (5)

Antes de proceder con la instalación del SO, se tienen que rellanar unos datos importantes como el nombre de usuario y la contraseña para acceder al sistema.

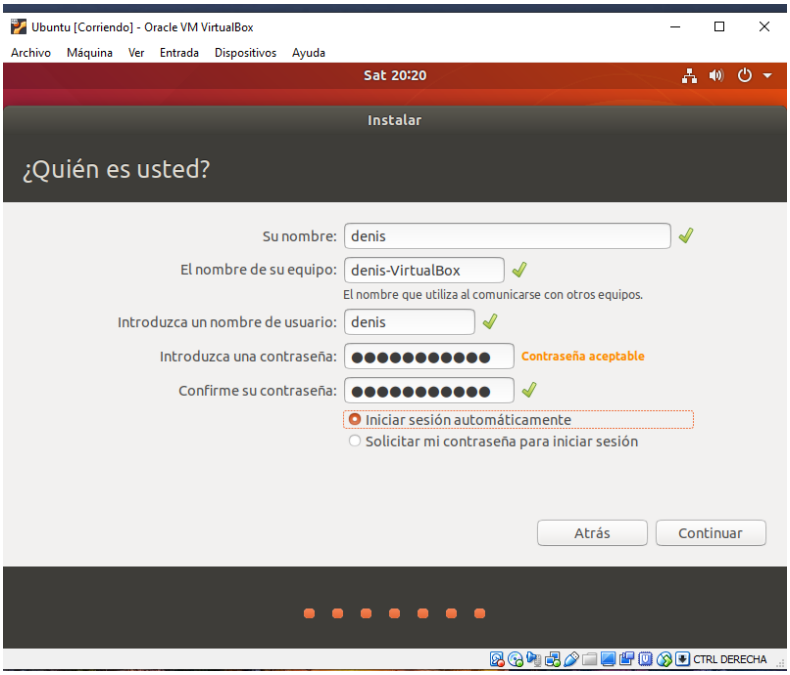

Figura 57 - Instalación de Ubuntu (6)

Una vez rellenados todos los datos, toca proceder a la instalación del sistema. Este proceso durará unos minutos.

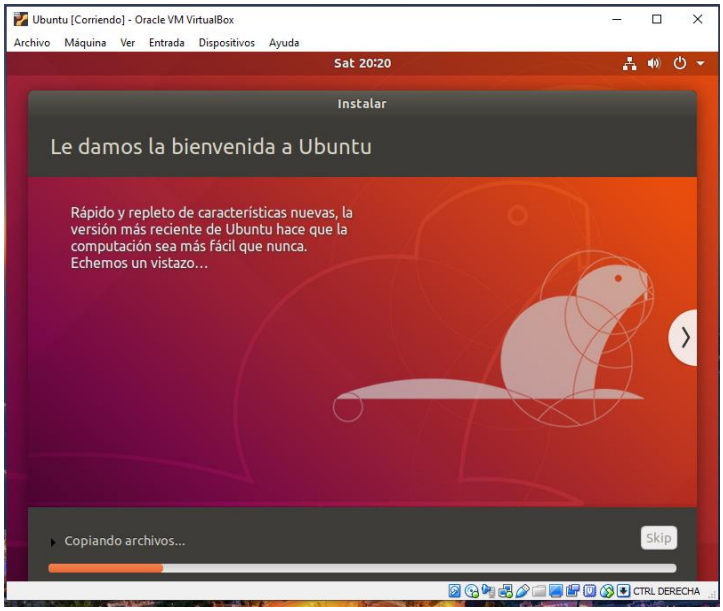

Figura 58 - Instalación de Ubuntu (7)

Instalación de Docker

Para instalar Docker en Ubuntu 18.04 LTS, se tienen que ejecutar los siguientes comandos (en orden):

```
$ sudo apt-get update

$ sudo apt-key adv --keyserver hkp://p80.pool.sks-keyservers.net:80 --recv-keys 58118E89F3A912897C070ADBF76221572C52609D

$ sudo apt-add-repository 'deb https://apt.dockerproject.org/repo ubuntu-xenial main'

$ sudo apt-get update

$ apt-cache policy docker-engine

$ sudo apt-get install -y docker-engine
```

Más información en la página oficial de Docker: https://docs.docker.com/install/linux/docker-ce/ubuntu/

Instalación de Docker-compose

Compose es un complemento de Docker que sirve para definir y ejecutar aplicaciones Docker multi-contenedor.

Para instalar Docker-compose en Ubuntu 18.04 hay que ejecutar los siguientes comandos (en orden):

```
$ sudo curl -L "https://github.com/docker/compose/releases/download/1.25.0/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose

$ sudo chmod +x /usr/local/bin/docker-compose

$ docker-compose --version
```

Más información en la página oficial de Docker: https://docs.docker.com/compose/install/

Instalación de NodeJS en Ubuntu

Para instalar NodeJS en Ubuntu 18.04 hay que ejecutar los siguientes comandos (en orden):

```
$ sudo apt-get install curl
$ curl -sL https://deb.nodesource.com/setup_12.x | sudo -E bash -
$ sudo apt-get install nodejs
```

Instalación de libindy en Ubuntu

Pre-requisitos:

- NodeJS
- Docker
- Docker-compose
- Git (ya viene instalado en Ubuntu 18.04)

Para instalar libindy en Ubuntu 18.04 hay que ejecutar los siguientes comandos (en orden):

```
$ sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys 68DB5E88
$ sudo add-apt-repository "deb https://repo.sovrin.org/sdk/deb xenial master"
$ sudo apt-get update
$ sudo apt-get install -y libindy
```

Más información en: https://medium.com/@spsingh559/hyperledger-indy-installation-with-ubuntu-and-node-js-4c1bff81ebc7

ANEXO II: Información adicional

Acciones en Alastria ID

En este apartado se describen todas las acciones que se pueden realizar en Alastria ID.

Cabe recordar que los nodos de Alastria son de tipo permisionado y por tanto no es posible realizar ninguna interacción con la cadena de bloques desde el exterior. Por ello se necesita un Gateway (GW) que será gestionada en combinación con los nodos y por los mismos actores. Los usuarios finales necesitan el apoyo de los socios que operan un nodo y la GW para todas las acciones.

A continuación, se describen paso a paso cada una de las acciones de Alastria ID según "Alastria identity core team". [57]

Creación de Alastria ID

1. Creación del par de claves (pública y privada) y, en su caso, el par de claves de dispositivo en un dispositivo con la app de Alastria instalada.
2. Identificación en los sistemas tradicionales (off-chain) del socio y selección de la opción de creación de identidad Alastria.
    a. Se crea un token de sesión que permite enlazar esta identificación con el siguiente paso.
    b. Se le solicita al usuario la clave pública de dispositivo.
3. Creación de Alastria ID, enviando al GW los datos siguientes:
    a. Token de sesión.
    b. Clave del dispositivo firmada con la clave personal.
4. El GW verifica que no haya otro Alastria ID para ese usuario (comprobación sobre la clave del dispositivo).
5. El GW genera una transacción de creación de Alastria ID:
    - De: Gateway.
    - Para: MetaIdentityManager.
    - Función: CreateIdentity(DeviceKey...)

6. El GW devuelve el alastria_id al sistema tradicional del socio, así como a la app móvil junto con otros parámetros necesarios de la red.

Autenticación con Alastria ID

1. Acceso a la WebApp y selección de Alastria ID como identificación. Inicialización de la sesión.
2. Se envía un mensaje de requerimiento de identificación al móvil, junto a un JWT firmado con:
    - Session Key.
    - Alastria ID de la aplicación.
    - Dirección callback del GW.
3. Solicitud de la clave pública de la aplicación.
4. Comprobación de la identidad de la aplicación por la Aplicación Alastria.
5. La Aplicación Alastria manda el visto bueno de la sesión, firmado por KPersonal, al gestor de sesiones.
    - Session Key.
6. Se recupera la KPub del usuario del Registry.
7. Se comprueba la firma del usuario y de su identidad.
8. Si es la primera vez que se accede con Alastria ID puede ser necesario enlazarlo con la cuenta en el sistema del proveedor de servicios:
    - Utilizando usuario/contraseña u otro sistema de autenticación.
    - Pidiendo atributos básicos (Nombre/apellidos, DNI…).
9. Envío del token de sesión JWT a la aplicación web.

Creación de testimonios

1. Identificación del usuario en el sistema tradicional del socio con uno de los siguientes mecanismos:
    - Token de sesión de Creación de Alastria ID.
    - Token de sesión de Autenticación con Alastria_id.
    - Credenciales del sistema tradicional del socio.

2. Creación de testimonios firmados para cada atributo en formato JWT y envío al móvil del usuario.
    - Consulta de datos validados del usuario en el sistema tradicional.
    - Generación del testimonio firmado en formato JWT para cada uno de los datos validados.
    - Envío de cada testimonio al móvil del usuario y almacenamiento en el repositorio accesible solo por el usuario.

Revocación de testimonios

Para revocar un testimonio es necesario remitir una transacción a la Blockchain de Alastria con los siguientes parámetros:

- De: Clave del revocador (emisor/usuario).
- Para: Registry.
- Función: Revoke (SHA-3 del testimonio).

Entrega de datos personales

1. Creación de la solicitud del solicitante de la identidad de:
    - La lista de atributos requeridos.
    - El callback donde deben ser enviados los datos.
    - Firma del requerimiento (a través de un JWT).
2. Envío de la solicitud de datos al usuario mediante:
    - URL inter-aplicación.
    - Notification al móvil.
    - Código QR para ser escaneado mediante la app móvil de Alastria.
3. La aplicación valida el requerimiento y realiza las siguientes acciones:
    - La aplicación selecciona los testimonios más adecuados.
    - Se los presenta al usuario para su validación.
    - Permite que el usuario cambie la elección predefinida.
    - Permite la aprobación o el rechazo de la solicitud.
4. Remisión del callback con la respuesta:

- Rechazo.
- Aprobación: con los testimonios elegidos por el usuario (incluyendo sus firmas originales) más el requerimiento original, en un JWT firmado con la clave privada del usuario.

Publicación de información pública del usuario

Clave personal pública del usuario en un repositorio

1. Se guarda la clave pública del usuario en un objeto JSON en IPFS.
2. El usuario registra mediante su alastria_id el hash IPFS de su clave pública personal en el Registry.

Publicación de Metadatos en el Registry

1. Se analiza la URI correspondiente al JWT de cada testimonio.
2. Registrar los atributos, enviando al GW los siguientes datos:
    a. JWT. El GW comprueba la firma del emisor.
    b. Transacción firmada.
        i. De: Usr (KPub cliente). El GW comprueba que la cuenta existe.
        ii. Para: IdentityManager. El GW comprueba la dirección.
        iii. Función: Forward. El GW comprueba la función llamada Destination = Registry.Set.

Transacciones

Cuando una aplicación Alastria desea que un usuario invoque un contrato:

1. La aplicación genera un JWT firmado con los datos de la transacción a firmar por el usuario.
    - Descripción sencilla de lo que se está solicitando.
    - Dirección del contrato.
    - Función a invocar en el contrato.
    - Valores de los parámetros.
    - Callback de respuesta.

2. La aplicación presenta dicha descripción y los datos de la transacción para su aceptación o rechazo.
3. En caso de aceptación, se firma la transacción y se remiten los datos al Gateway para su envío a la Blockchain.
4. El Gateway envía el hash de la transacción al móvil.
5. El móvil envía a la URL de callback el hash de la transacción.

Recuperación de claves privadas

1. Identificación ante los sistemas tradicionales (off-chain) del socio e inicialización del proceso de recuperación de claves:
    a. Se genera un token de sesión que permite enlazar esta identificación con el paso siguiente.
    b. Se entrega el Alastria ID correspondiente al usuario identificado.
    c. Se le solicita al usuario la llave pública de dispositivo.
2. Inicio de la recuperación de clave, enviando al GW los siguientes datos:
    a. Token de sesión.
    b. Llave de dispositivo firmada por la clave de dispositivo (el usuario no posee su clave personal).
3. El GW verifica que no haya otro Alastria ID para ese dispositivo.
4. El GW solicita al sistema tradicional la parte de la clave personal Alastria correspondiente al Alastria ID y el token de sesión.
5. Se repite el proceso hasta obtener n-1 claves.
6. Con las n-1 partes se recompondrá y descifrará el par de claves.

Acciones en uPort

En este apartado se describe paso a paso todo el proceso necesario para realizar alguna acción en uPort (firma de transacciones, recuperación de claves, etc.).

Firmar transacciones

La firma de transacciones en uPort contiene los 10 pasos siguientes [44]:

1. El navegador muestra el código QR con el ID de sesión generado aleatoriamente en una URI.
2. El navegador comienza a sondear (*polling*[4]) el servidor Chasqui usando el ID de sesión para comprobar si el móvil ha contabilizado el hash de la transacción.
3. Si la comunicación es de móvil a móvil, se sigue el patrón de interacción móvil (usando URIs y JWTs en lugar del servidor Chasqui, tal y como se ve en la figura 18).
4. El móvil escanea el código QR y muestra una ventana que le pide al usuario firmar una transacción. Se muestra un cuadro de diálogo del sistema (Touch ID / Face ID / PIN del dispositivo) para acceder a la clave del dispositivo.
5. Si el usuario da su consentimiento: el móvil obtiene los datos de la transacción de la URL, así como el ID de sesión, y después firma los datos de la transacción con la clave del dispositivo.
6. El dispositivo envía la transacción firmada al servidor *sensui.uport.me*. Sensui añade la transacción firmada en una nueva transacción y la envía a un contrato de retransmisión a través de *rinkeby.infura.io*.
7. Infura devuelve el hash de la transacción a Sensui, y éste lo pasa a la aplicación móvil. El móvil envía el hash de la transacción a Chasqui utilizando el ID de sesión.
8. El navegador recibe el hash de la transacción de Chasqui, elimina el código QR de la interfaz de usuario y actualiza la interfaz de usuario en consecuencia.
9. El contrato transmisor confirma que la transacción original firmada no ha sido manipulada, y luego envía la transacción completa a través del contrato controlador al contrato proxy.
10. El contrato proxy reenvía la transacción a su destino. El gas de dicha transacción siendo pagado por el servidor Sensui.

[4] *Polling - Es el sondeo que realiza un servidor para comprobar el estado de cada terminal en una red.* [49]

A continuación, se muestra un diagrama en donde se puede visualizar todo el proceso descrito anteriormente:

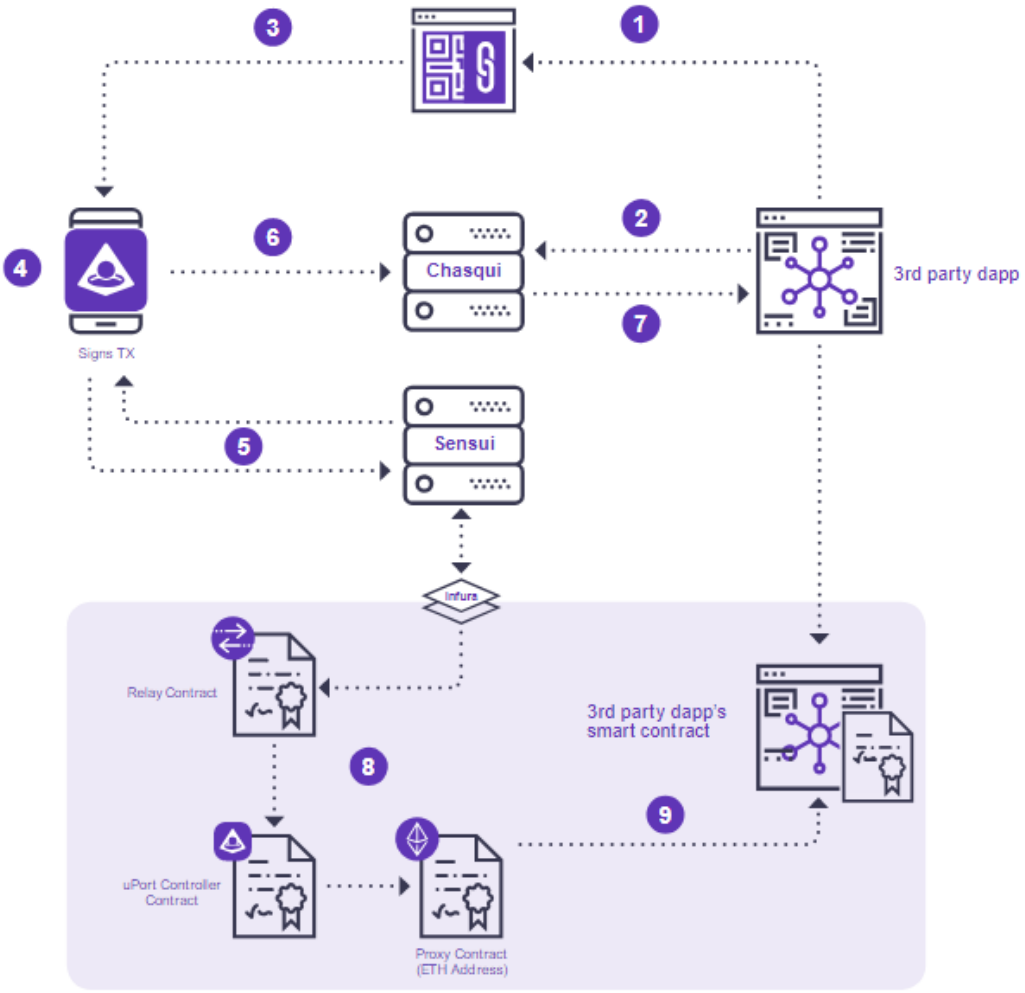

Figura 62 - Proceso de firma de transacciones en uPort [48]

Solicitud de credenciales

La solicitud de credenciales en uPort se lleva a cabo mediante los cinco pasos siguientes [44]:

1. El navegador muestra el código QR con la URI

2. El navegador sondea (polling) al servidor Chasqui usando el ID de sesión para comprobar si el móvil ha publicado la dirección y cualquier otra información requerida por la aplicación de terceros.
3. El móvil escanea el código QR y le muestra al usuario un aviso para que comparta su dirección (y, opcionalmente, otros datos relevantes).
4. Si el usuario da su consentimiento, el móvil toma el ID de sesión de la URI y envía la dirección y los datos a la API de Chasqui usando dicho ID de sesión.
5. El navegador recoge la dirección y los datos de Chasqui y elimina el código QR de la interfaz de usuario.

A continuación, se muestra un diagrama en donde se puede visualizar todo el proceso descrito anteriormente:

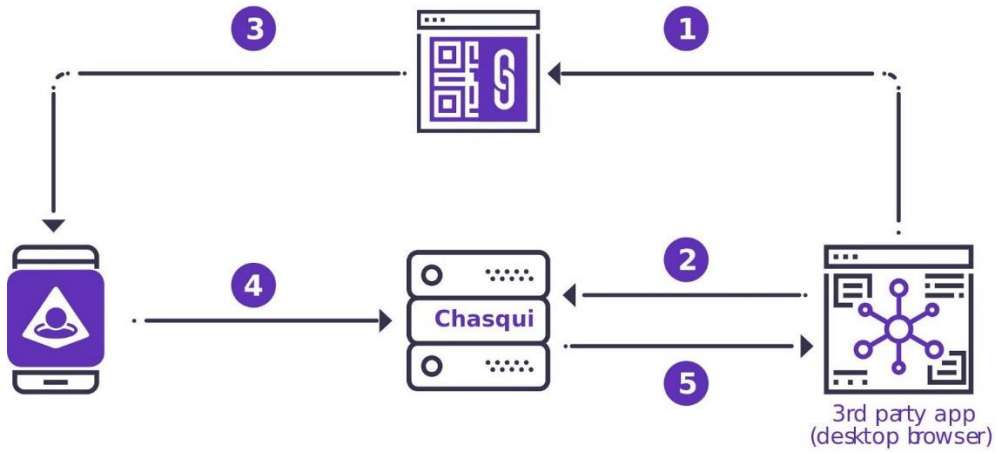

Figura 63 - Solicitud de credenciales en uPort [48]

Certificación (atestado) de credenciales

La certificación de credenciales en uPort se lleva a cabo mediante los pasos expuestos a continuación [44]:

1. El navegador muestra un código QR (si se trata de la versión de escritorio) o carga una URI que abre la aplicación uPort (si es la versión móvil) para iniciar el flujo de datos de inicio de sesión o compartir el flujo de datos indicado en la solicitud de credenciales.
2. Después de que el usuario escanee el código QR (en el escritorio) o dé su consentimiento para abrir la aplicación (en el móvil), la aplicación muestra un aviso pidiendo al usuario

que comparta sus datos. Este aviso contendrá la dirección del usuario y también podrá contener un "*push token*", así como cualquier otro dato que la aplicación decida solicitar.

3. Si el usuario da su consentimiento, la aplicación móvil publica la dirección a través del servidor Chasqui (si es la versión escritorio) o codificado en un JWT adjunto a una URI (si es la versión móvil).
4. El navegador recoge la dirección y los datos descritos en el paso anterior y elimina el código QR de la interfaz de usuario.
5. Cuando la aplicación está lista para enviar una certificación, codifica los datos relevantes en un JWT y lo firma.

- Si el modo "*push*" está habilitado:
    - 6a. La certificación se envía junto con el token push del usuario a *Pututu*[4].
    - 7a. Pututu comprueba la firma del token push contra la clave pública del usuario y luego reenvía el token de certificación.
    - 8a. El usuario recibe una notificación push, que genera un aviso en la aplicación preguntando si desea aceptar la certificación.
- Si no está activado el modo "*push*":
    - 6b. El token de certificación se codifica en un código QR (en la versión de escritorio) o en una URI (en la versión móvil) y se le indica al usuario que escanee o abra la app de uPort, respectivamente.
    - 7b. Después de que el usuario escanee (escritorio) o abra la app (móvil), se muestra un aviso preguntando si el usuario desea aceptar la certificación.

[5] *Pututu es un servidor que permite a las aplicaciones basadas en Blockchain (dApps) y a los servidores enviar mensajes push a cualquier aplicación de uPort Mobile*. [50]

A continuación, se muestra un diagrama en donde se puede visualizar todo el proceso descrito anteriormente:

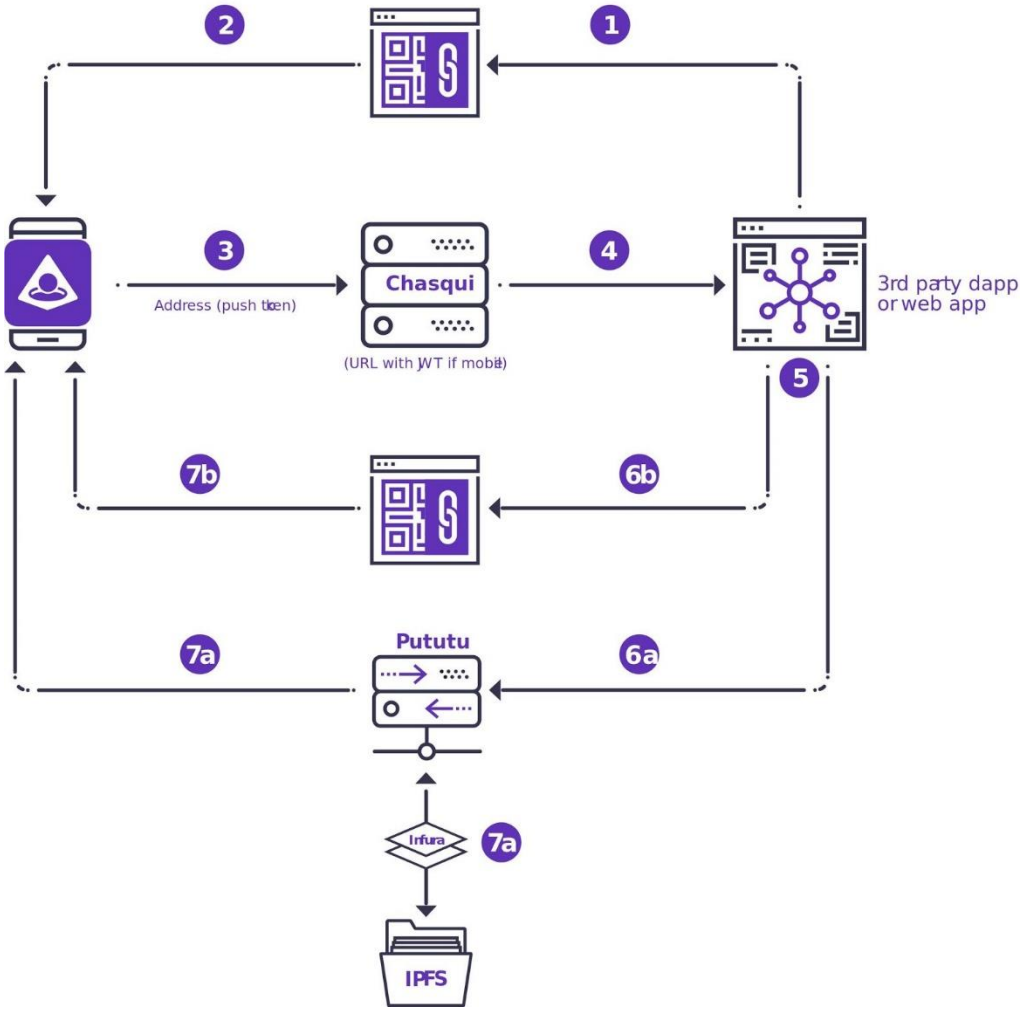

Figura 64 - Funcionamiento del proceso de certificación en uPort [48]

Código de la aplicación

Tal y como se ha explicado en el apartado número 9.1 “Trabajo futuro”, la aplicación desarrollada tiene un amplio potencial de crecimiento y de mejoras. Por lo tanto, se pretende que la aplicación sea accesible por cualquier persona. Debido a esto, la aplicación está disponible como software libre, bajo una licencia abierta, que permite que cualquier usuario pueda tanto analizar el código y entender cómo funciona como disponer del código para, en base a él, desarrollar sus propias soluciones.

El código de las dos aplicaciones se encuentra disponible en los siguientes enlaces:

UOC ID

https://github.com/denisro2010/UOC_ID

MED ID

https://github.com/denisro2010/MED_ID