UNIVERSIDAD DE MÁLAGA

E.T.S. INGENIERÍA INFORMÁTICA
UNIVERSIDAD DE MÁLAGA

Grado en Ingeniería Informática

La identidad autosoberana como solución descentralizada a la custodia de credenciales privadas

Self-Sovereign Identity as a decentralized solution to the custody of private credentials

Realizado por
Marcos Hidalgo Baños

Tutorizado por
Isaac Agudo Ruiz
Rodrigo Román Castro

Departamento
Lenguajes y Ciencias de la Computación

MÁLAGA, septiembre de 2024

UNIVERSIDAD DE MÁLAGA

E.T.S. INGENIERÍA INFORMÁTICA
UNIVERSIDAD DE MÁLAGA

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA

GRADUADO EN INGENIERÍA INFORMÁTICA

La identidad autosoberana como solución descentralizada a la custodia de credenciales privadas

Self-Sovereign Identity as a decentralized solution to the custody of private credentials

Realizado por
Marcos Hidalgo Baños

Tutorizado por
Isaac Agudo Ruiz
Rodrigo Román Castro

Departamento
Lenguajes y Ciencias de la Computación

UNIVERSIDAD DE MÁLAGA
MÁLAGA, SEPTIEMBRE DE 2024

Fecha defensa: septiembre de 2024

Resumen

La Identidad Autosoberana plantea una nueva alternativa a la gestión de las credenciales que permiten identificar a sus participantes en entornos digitales. Para ello se basa en determinados mecanismos técnicos que garantizan la seguridad criptográfica, en innovadoras tecnologías descentralizadoras fomentando así la transparencia en el sistema y en crear un ecosistema donde cada actor puede efectuar sus funciones de manera autónoma e independiente.

Mediante este TFG se pretende ofrecer unas correctas y actualizadas definiciones de dichos aspectos, las cuales servirán para elaborar un profundo análisis de aplicabilidad del paradigma en un entorno de Gemelo Digital. Además, se introducirá el plan de progresiva implantación del mismo a la sociedad europea con la Identidad Digital Europea, se realizará una comparativa con otros entornos como el del Internet de las Cosas y se desarrollará una prueba de concepto que ilustrará el funcionamiento de la tecnología desde un enfoque práctico.

Finalmente, se presentarán a modo de conclusión un conjunto de reflexiones personales fruto de la labor de investigación, donde también se expondrán nuevas ideas y valoraciones que complementarán el análisis llevado a cabo.

Listado de palabras clave:

- **Identidad Autosoberana.**
- **Gemelos Digitales.**
- **Blockchain.**
- **Identidad Digital Europea.**
- **Internet de las Cosas.**

Abstract

The Self-Sovereign Identity presents a new alternative for managing credentials that allow the identification of its participants in digital environments. To do so, it relies on certain technical mechanisms to ensure cryptographic security, innovative decentralizing technologies that promote transparency in the system, and the creation of an ecosystem where each actor can perform their functions in a autonomous and independent way.

This Bachelor Thesis aims to offer correct and updated definitions of these aspects, which will be used to develop a deep analysis of the paradigm's applicability in a Digital Twins environment. Additionally, we will introduce the plan for its progressive implementation in the european society through the European Digital Identity, a comparison with other environments such as the IoT, and a proof of concept will be developed for illustrating the technology funcionalities with a practical approach.

Finally, a set of personal reflections resulting from the research work will be presented as a general conclusion, including new ideas and valorations that will complement the analysis.

Keywords:

- **Self-Sovereign Identity.**
- **Digital Twins.**
- **Blockchain.**
- **European Digital Identity.**
- **Internet of Things.**

Índice

1 Introducción

1.1. Objetivos.

La razón de ser de este trabajo fin de grado es dual puesto que se persigue realizar un balance entre los aspectos técnicos y legales del paradigma de la Identidad Autosoberana (SSI), del término inglés ”Self-Sovereign Identity”, a nivel tanto de individuos como de activos digitales, teniendo en cuenta cómo se planea implementar a nivel europeo la Identidad Digital Europea (eIDAS) cuyo objetivo es resolver esta misma problemática.

Por otra parte, se persigue estudiar la aplicación del paradigma en la autenticación y autorización de componentes y activos digitales en un entorno de Gemelo Digital (DT), así como la identificación de los posibles problemas que el uso de dicha tecnología podría acarrear. Precisamente, mediante la realización de una prueba de concepto se verificará la viabilidad de la tecnología empleada, sacando a la luz los puntos fuertes y las deficiencias de la misma.

1.2. Motivación.

La identidad autosoberana proporciona a los individuos control sobre sus propias credenciales, de manera que una persona sea capaz de autenticarse ante cualquier servicio (p.ej. un comercio web) sin que este tenga que guardar información que identifique unívocamente a dicha persona (p.ej. contraseña del usuario) o que dependa de la verificación por parte de un agente externo (tercero de confianza) que lo garantice.

Este paradigma y sus tecnologías subyacentes, como pueden ser las carteras digitales para la gestión de credenciales o las cadenas de bloques para garantizar la descentralización, aún no han sido del todo generalizadas para el gran público. Sin embargo, son numerosas las entidades públicas y privadas que apuestan por implementarlas como solución a las carencias detectadas durante la pandemia y como manera de adaptarse al cambiante entorno.

Un claro ejemplo de ello es la normativa europea que planea establecer este paradigma a todos sus ciudadanos. Esta generalización hacia las necesidades del gran público va más allá de la identificación de individuos, puesto que la SSI también podría aplicarse a objetos y/o servicios como entidades de la Internet de las Cosas (IoT).

No obstante, existe la necesidad de realizar una mayor investigación y documentación en este campo, especialmente en áreas como los Gemelos Digitales (uno de los pilares del ecosistema industrial conocido como la Industria 5.0). Es necesario entonces analizar sus aspectos técnicos para realizar una valoración de la situación actual en el ámbito general.

1.3. Estructura del documento.

Para lograr una correcta organización de los contenidos expuestos, se decide clasificar por secciones según el tema central a definir o analizar. De tal manera, podemos encontrar:

- Sección 2, Estado del arte.

  El escrito comenzará mediante un breve repaso de la historia de la Web desde sus orígenes y su enfoque con respecto a la gestión de las credenciales. A continuación se expondrán los aspectos teóricos, actualizados a fecha de redacción.

- Sección 3, Aplicabilidad de la Identidad Autosoberana.

  En esta sección analizaremos de forma teórica de qué forma la identidad autosoberana puede cumplir con los requisitos y necesidades de un entorno de gemelos digitales.

- Sección 3.3, Prueba de concepto.

  Por otro lado, elaboraremos una prueba de concepto que nos permita poner en práctica lo estudiado del paradigma, siendo posible así conocer los aspectos técnicos del mismo.

- Sección 4, Conclusiones.

  Finalmente se procederá a valorar el paradigma, a la vez que se medirán las similitudes y diferencias con respecto a lo propuesto por la normativa europea.

1.4. Metodología de trabajo.

Debido a la naturaleza del trabajo (investigación realizada en solitario) la metodología empleada pretende ser capaz de diseñar un **estudio** que genere resultados adecuados para realizar una **valoración** a posteriori. Por lo tanto, este proceso deberá ser iniciado por una intensa fase de recopilación de información encargada de establecer las bases iniciales.

Una vez concluida, se podrá iniciar en paralelo cualquier otro aspecto relacionado con el objetivo dual del proyecto, tanto el técnico con las pruebas de concepto como el legal con la normativa europea, a su vez, deberá ser **flexible** y así poder aceptar cambios ocasionados por nuevas fuentes o ideas obtenidas durante su transcurso (retroalimentación).

Esto generará fragmentos con conclusiones y valoraciones que deberán ser organizados durante la fase final de la redacción definitiva de la memoria y presentación. A modo de listado, se procede a describir las diferentes etapas del mismo:

1. **Fase de investigación (70h)**. Obtención de información sobre los aspectos técnicos y situación actual del establecimiento del paradigma, tanto a efectos teóricos como prácticos. Esto dará lugar a ideas para ser incluidas en posteriores etapas.

2. **Creación de las pruebas de concepto (60h)**. Tras haber analizado las bases técnicas sobre las que se sustenta, comprobar su viabilidad a la hora de implementar.

3. **Comparativa con la normativa europea (60h)**. Analizar el plan de establecimiento de este paradigma, con sus similitudes y diferencias a lo teóricamente estudiado.

4. **Valoración general (45h)**. Balance de ventajas e inconvenientes como consecuencia de optar por esta solución.

5. **Redacción de la memoria (45h)**. Una vez concluido el desarrollo del estudio, estructurar y argumentar los resultados obtenidos. Documentar el proceso llevado a cabo, cambios durante su transcurso, referencias empleadas, etc.

6. **Elaboración de la presentación (16h)**. Exposición de los contenidos de la memoria en forma de presentación como recurso empleado para la defensa.

2

Estado del arte

2.1. Evolución de la Web.

En esta sección se tratará de introducir y contextualizar las principales características que han definido a cada una de las etapas de la Web a lo largo del tiempo. Este análisis previo se centrará exclusivamente en la gestión del **almacenamiento de las credenciales** que identificarán a los usuarios, puesto que es el tema que nos atañe en este trabajo.

Figura 1: *Gestión de las credenciales en cada etapa.* Fuente [5]

Durante su primera etapa, en la denominada como **Web 1.0** la autenticación no era una de las principales preocupaciones puesto que internet tenía fines informativos [1]. Los usuarios accedían a los recursos web como si de un tablón de anuncios se tratase, por lo que la información debía ser de acceso libre y los sitios eran estáticos con apenas interactividad.

Sin embargo, los datos alojados en ella estaban siendo almacenados localmente en bases de datos distintas para cada servicio. Este hecho supuso un problema de **dispersión** y **duplicidad** de información a lo largo de la red que se vio enfatizado a medida que se introducían las primeras credenciales de usuario, guardándose además en **texto plano**.

Una vez se asentó la necesidad de identificar a los usuarios que accedían a los recursos web, dio comienzo la **Web 2.0** y con ella la introducción del dinamismo y el contenido generado por los propios usuarios. En lo referente a la seguridad, se optaron por algoritmos criptográficos Hash para el almacenamiento de las credenciales y protocolos para el establecimiento seguro de la conexión como HTTPS, lo cual supuso una mejora sustancial a la anterior etapa.

No obstante, el problema de la dispersión y duplicidad de los datos seguía presente. De hecho, esto afectaba en gran medida al nuevo paradigma de las credenciales, puesto que nuevas soluciones basadas en la **confianza con terceros** ayudaron a la centralización de los datos.

Reflexión

No es de extrañar que la segunda versión de la Web fuera una expansión de la primera que tratase de solventar los problemas originados por la misma, viéndose acentuados por la masiva adopción. Entonces, ¿ya hemos terminado? ¿Es lo mejor que podemos conseguir o existe algún aspecto con margen de mejora?

Actualmente nos encontramos en proceso de transición a la **Web 3.0** la cual promete traer un paradigma **descentralizado** (sin dependencia de terceras partes) aún manteniendo las bases construidas en las anteriores etapas. Para ello propone introducir nuevas tecnologías como las Carteras de Identidad Digital, el Identificador Descentralizado (DID) y la Credencial Verificable (VC) que mejorarán la seguridad y privacidad de los usuarios, a la vez que permitirá a las personas recuperar el control sobre sus propios datos personales [1].

2.2. Identidad Autosoberana.

2.2.1. Definición del concepto.

Una primera manera de concebir la Identidad Autosoberana (SSI) como alternativa a la tradicional manera de gestión de credenciales es percatándonos del papel que el propio usuario tiene tras el establecimiento de este paradigma. Mientras que normalmente no disponían de métodos que de manera autónoma pudieran identificarlos, haciéndolos dependientes de una tercera parte, ahora acceden a los diferentes servicios identificándose ellos mismos mediante credenciales autogestionadas en sus carteras digitales.

Encontramos definiciones similares en esta misma línea leyendo la literatura, como es el caso de [12] donde se nos muestra que "la Identidad Autosoberana es una representación digital de las características, descripciones e identificadores de los individuos donde ningún gobierno u organización puede violar nuestro derecho a elegir nuestro nivel de privacidad o grado de visibilidad con nuestros atributos de identidad."

Figura 2: *Comparativa entre ambos paradigmas.* Fuente [11]

Este radical cambio necesariamente requiere redefinir aspectos clave como la función que desempeñan los diferentes actores y la propia arquitectura de sus componentes, además de dotar con medios técnicos a los usuarios.

2.2.2. Actores y sus funciones en el paradigma.

Tal y como se detalla en [13], nos encontramos en un escenario con tres principales actores:

- **Emisor** (Issuer). Responsable de **generar** las credenciales pedidas por los usuarios. Pueden ser entidades gubernamentales, corporaciones e incluso individuos.
- **Titular** (Holder). Es el usuario propiamente dicho, el cual **dispone** de sus credenciales expedidas por el emisor y las almacena en su Cartera de Identidad Digital.
  Representa a todo aquel portador de Identidad Digital, de cualquier naturaleza.
- **Verificador** (Verifier). Recibe y procesa las credenciales para **verificar** su validez.
  Representan a aquellos interesados en comprobar la identidad de los Titulares, como pueden ser las instituciones o los negocios.

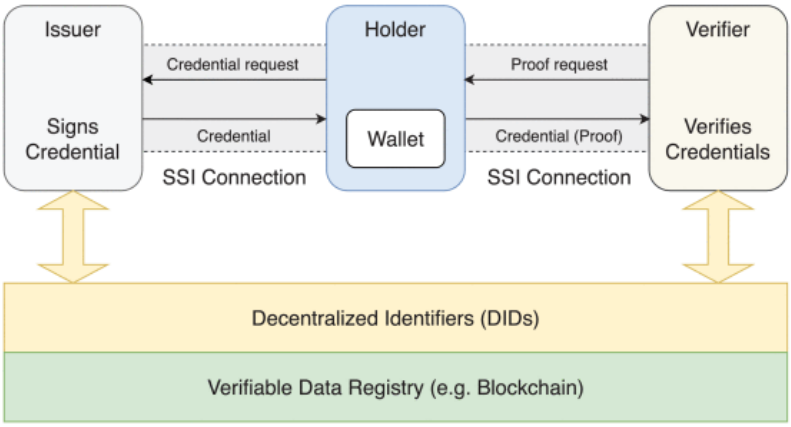

Figura 3: *Principales actores de la SSI y sus comunicaciones.* Fuente [13]

Observación 1

Es importante destacar que se mediante este esquema se desacopla la emisión y verificación de credenciales, del registro de estas. De esta manera se permite desplegar los DIDs sobre una blockchain o sobre un repositorio centralizado indistintamente.

2.2.3. Arquitectura y marco de referencia.

También en [13] se elabora una **arquitectura multicapa** con el objetivo de estandarizar e "integrar la tecnología con la responsabilidad humana en todas las capas legales y sociales". Destacar que existen diversos marcos de referencia expuestos en numerosos artículos académicos, pero se decide destacar este en concreto ya que realiza una división en pilas (stacks) para estructurar sus componentes, los cuales varían entre **fundamentos técnicos** en capas inferiores y **protocolos** en las superiores.

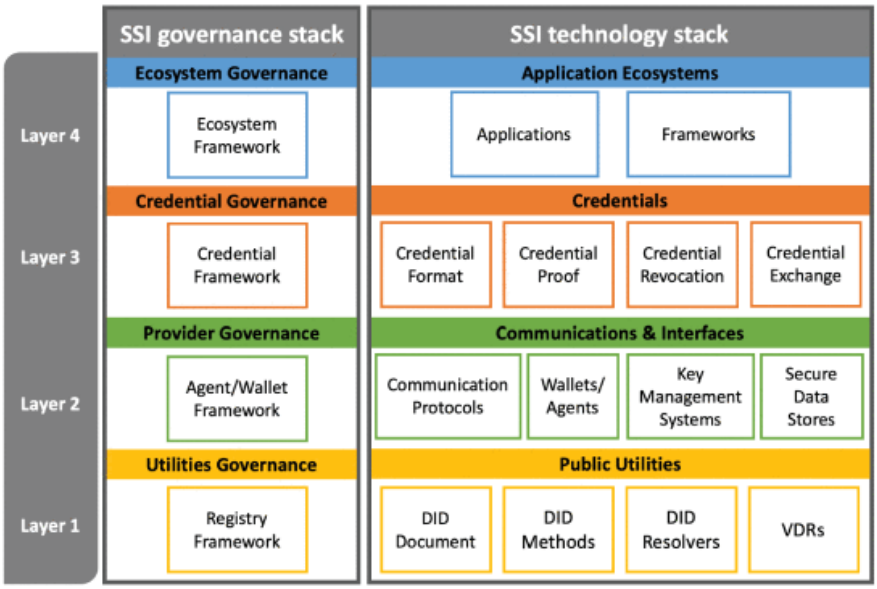

Figura 4: *Arquitectura multi-stack para la SSI.* Fuente [13]

Esta arquitectura propuesta resulta interesante porque tiene en cuenta esta **dualidad** en el paradigma, siendo de gran utilidad en secciones posteriores. Además, aunque no se entra a describirla en profundidad en este TFG, nos ayuda a comprender con mayor facilidad los fundamentos técnicos sobre los que se sustenta.

En la misma línea, se recomienda la lectura de [12] en la que se propone un marco de evaluación para los sistemas de la SSI y se realiza un análisis aplicándolo a uPort, Sovrin, ShoCard, Civic y Blockstack, las cuales representan diferentes **implementaciones** de este paradigma.

A continuación, se detallaran las tecnologías que permiten que este paradigma sea factible. La principal fuente de información de esta sección es World Wide Web Consortium (W3C), aunque será complementada por otras alternativas en determinadas subsecciones.

A fecha de la elaboración de este escrito, las versiones más actualizadas corresponden a "Decentralized Identifiers (DIDs) v1.0" [3] y "Verifiable Credentials Data Model v2.0" [15].

2.2.4. Carteras de Identidad Digital.

Seguramente sea el componente mayormente reconocido del paradigma, debido a que resulta familiar al usuario (paralelismo con la cartera física) y es de uso generalizado en otras tecnologías como las Criptodivisas o métodos de pago alternativos como Google Wallet.

Sin embargo, esta cartera 'genérica' debe ser adaptada al paradigma para que sea capaz de "firmar, cifrar, reenviar mensajes relacionados con credenciales y establecer conexiones de agente a agente" [13], además de permitir el almacenamiento de las propias credenciales.

Observación 2

El hecho de que una Cartera Digital (o solución que la implemente) pueda almacenar documentos identificativos como pasaportes, DNIs o similares **NO** quiere decir que sea una Cartera de Identidad Digital, al menos no como la concebimos en este escrito.
Recordar que deben cumplirse los requisitos técnicos que garanticen la descentralización e implementación de funcionalidades del paradigma de la SSI, entre otros aspectos.

2.2.5. Identificadores Descentralizados.

Como hemos visto anteriormente, la base técnica sobre la que se sustenta el paradigma es el Identificador Descentralizado (DID). Estos siguen el formato de un Identificador Uniforme de Recursos y están compuestos por tres elementos:

- **Esquema** (Scheme).
  Definido como 'did', representa al esquema de identificación a seguir.
- **Método DID** (DID Method).
  Indica la manera de realizar la resolución del DIDs. Destacar que algunas de estas implementaciones fueron ya mencionadas en 2.2.3. Por ejemplo, 'sov' correponde a Sovrin.
- **Identificador Específico** (Specific Identifier).
  Permite asociar inequívocamente al usuario dentro del método en cuestión, al ser único.

Figura 5: *Estructura genérica de un DID.* Fuente [3]

A continuación, definiremos el resto de componentes que conforman la arquitectura sin entrar en profundidad en los detalles concretos de la implementación, como los parámetros que poseen o interfaces disponibles.

- **URL DID** (DID URL).
  Son una extensión sintáctica de los DID y permite la referenciación a recursos internos o externos. Siguiendo el ejemplo, *did:example:123456789abcdefghi/path/to/rsrc* busca en el directorio descrito un archivo del paradigma, como los Documentos DID.
- **Documento DID** (DID Document).
  Es una representación de la información que describe a un Sujeto DID, el cual está siendo identificado mediante su DID, incluyendo mecanismos como las claves públicas.

- **Sujeto DID** (DID Subject).

  Corresponde con el usuario en sí mismo, independientemente de su naturaleza. Puede darse el caso en el que el propio sujeto sea a la vez Controlador DID.

- **Controlador DID** (DID Controller).

  Entidad o entidades capaces de modificar el contenido de un Documento DID, habiendo sido previamente autorizado por el Sujeto DID correspondiente.

- **Registro de Datos Verificables** (Verifiable Data Registry).

  Es el sistema de almacenamiento de DIDs para facilitar la creación de Documentos DID. Pueden tratarse de 'distributed ledgers', sistemas de archivos descentralizados, bases de datos de cualquier tipo, redes peer-to-peer y otras formas de almacenamiento de datos confiable [3]. Resulta interesante concebir el Registro de Datos Verificables (VDR) como la manera de "establecer confianza técnica y facilitar las interacciones entre los actores, los DIDs públicos y sus correspondientes Documentos DID" [13].

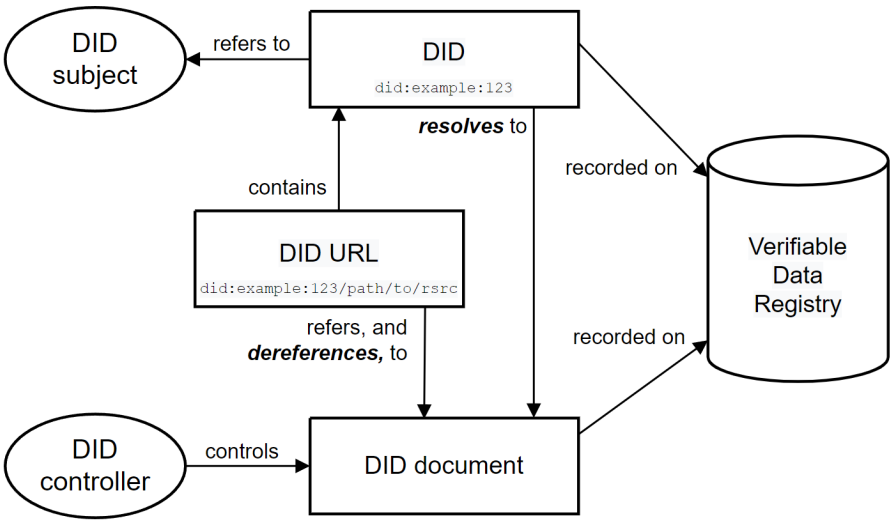

Figura 6: *Arquitectura inicial y relación entre componentes de los DIDs.* Fuente [3]

Partiendo de esta arquitectura en la que se establecen las bases teóricas del paradigma, podemos extenderla hasta incluir nuevos componentes y mecanismos que permitan la implementación de nuevas utilidades para la gestión de los DIDs y los Documentos DID.

- **Método DID** (DID Method).

  Asociado a una determinada VDR, define los mecanismos para crear, resolver, actualizar y revocar DIDs al igual que Documentos DID. Es el único componente capaz de modificar los contenidos de el VDR y por ello deben existir las diferentes implementaciones de las funciones de Resolución DID para al menos un Método DID.

- **Solucionador DID** (DID Resolver).

  Es capaz de generar un Documento DID a partir su correspondiente DID mediante la implementación de las funciones de Resolución DID.

  Un buen ejemplo de ello es la función de deferencia (deference). Mediande ellas se crea el Desreferenciador de URL DID (DID URL Dereferencer), que es un sistema capaz de obtener los recursos a partir de las URL DID.

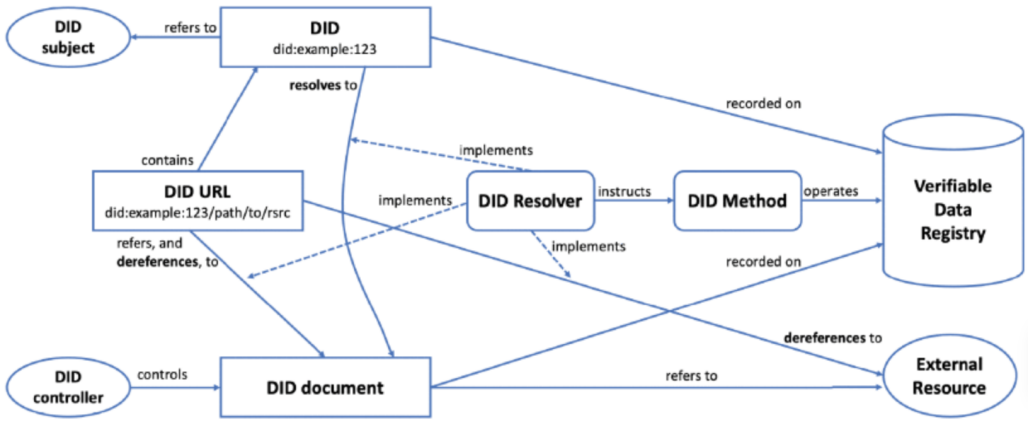

Figura 7: *Arquitectura extendida y relación entre componentes de los DIDs.* Fuente [13]

Con esta nueva arquitectura se consigue establecer una base técnica funcional que garantiza la identificación de sujetos de manera descentralizada. Sin embargo, el Identificador Descentralizado (DID) es empleado principalmente como un componente más en sistemas de mayor envergadura , como es el caso de la Credencial Verificable (VC).

2.2.6. Credenciales Verificables.

De igual manera que si de una credencial física se tratase, la Credencial Verificable (VC) permite al usuario identificarse digitalmente ya que contiene su información personal. Sin embargo, mediante el cifrado de estos datos empleando técnicas criptológicas como la Firma digital, se consigue que sea "verificable". Esto es debido a que cualquier verificador tiene la capacidad de comprobarlas de manera autónoma e independiente [15].

Como ya se habrá podido intuir, aspectos como los actores que participan en este modelo o los mecanismos que permiten la identificación descentralizada son los mismos que han sido descritos con anterioridad. No obstante, se introducen nuevos conceptos como los siguientes:

- **Afirmaciones** (Claims).
  Relación entre el sujeto sobre el valor asignado mediante la propiedad que se afirma. Por ejemplo, el autor de este escrito (sujeto) es alumno (propiedad) de la UMA (valor).
- **Credenciales** (Credentials).
  Conjunto de afirmaciones realizados sobre una misma entidad. Opcionalmente puede contener metadatos o pruebas (proofs) y se consideraría "verificable" si se realiza un cifrado de estos datos que permita probar criptológicamente quién es el emisor.
- **Presentaciones** (Presentations).
  Selección de determinadas VCs de un mismo Titular para facilitar su transmisión de forma minimalista, consiguiendo así compartir la menor información necesaria.

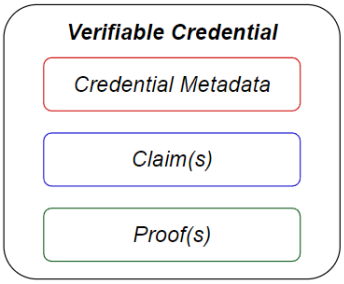

Figura 8: *Componentes de una VC*. Fuente [15]

2.3. Identidad Digital Europea.

2.3.1. Definición del concepto.

La denominada como Identidad Digital Europea (eIDAS) es el intento de la Unión Europea de establecer el paradigma de la Identidad Autosoberana (SSI) a todos sus ciudadanos, residentes y empresas, para que de tal manera "puedan beneficiarse de una cartera de identidad digital europea personal a partir de 2026" [6].

Entre los servicios y herramientas que se pretenden poder utilizar destacan los siguientes: "Firma electrónica, Sello de tiempo, Identificación electrónica, Certificado cualificado de autenticación de sitio web, Sello electrónico y Servicio de entrega electrónica certificada" [8].

Para ello, el denominado como **eIDAS Bridge** asistirá a los Emisores ayudando durante el firmado y a los Verificadores mediante la automatización del proceso de identificación de la organización tras el DID del Emisor [14]. Podemos entenderlo como una manera de comprobar si una determinada VC tiene validez (sea de confianza) para ser empleada en la Unión Europea.

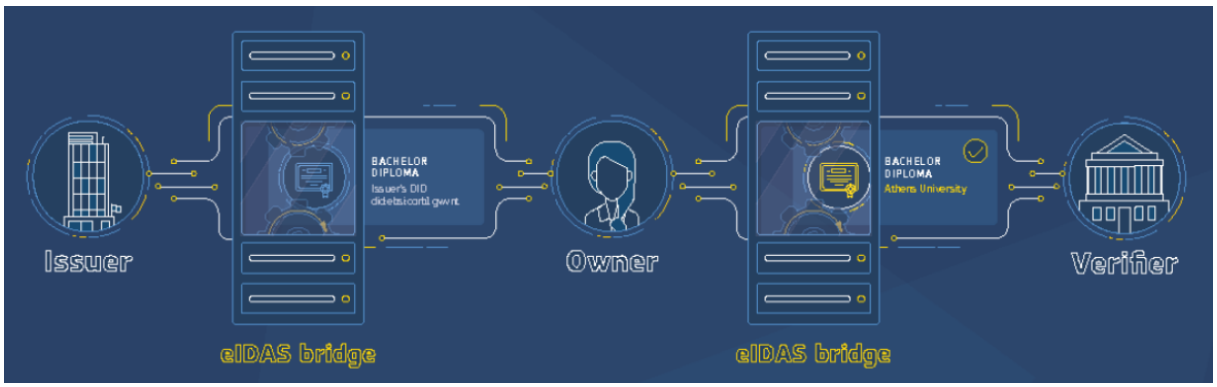

Figura 9: *Modo de funcionamiento del eIDAS Bridge.* Fuente [14]

2.3.2. Infraestructura Europea de Servicios Blockchain.

El anteriormente mencionado **eIDAS Bridge** forma parte del Marco Europeo de Identidad Soberana (ESSIF) y es uno de los principales casos de uso de la Infraestructura Europea de Servicios Blockchain (EBSI) [14], la cual es una iniciativa de la Comisión Europea y de la Asociación Europea de Blockchain para la implantación del paradigma al gran público.

Analizando lo propuesto, vemos como se mantienen aspectos fundamentales descritos con anterioridad. Sin embargo, es ahora la EBSI quién se encarga de cumplimentar el rol de VDR, tal y como se describe en su **Marco de Credenciales Verificables** [5]. Por lo tanto, la principal función de la organización es la de registrar en una base de datos, basada en Blockchain pública y por lo tanto descentralizada, las transacciones realizadas en el proceso de expedición y verificación de credenciales verificables.

Figura 10: *Papel de la EBSI en el paradigma.* Fuente [5]

Observación 3

Destacar la distinción que se realiza en [16] entre 'persona natural' y 'entidad legal', donde podemos encontrar información detallada de la adecuación del paradigma al ámbito legal europeo a causa del Reglamento General de Protección de Datos (GDPR).

2.3.3. Cartera de Identidad Digital Europea.

Actualmente, uno de los principales esfuerzos que se está llevando a cabo es el desarrollo de la Cartera de Identidad Digital Europea (EUDI Wallet), la cual ya se está probando en entornos reales mediante cuatro proyectos a gran escala que se lanzaron el 1 de abril de 2023 [8].

La principal fuente de información sobre los detalles de su implementación es [7], donde se puede encontrar el repositorio informativo raíz de la iniciativa en Github. Allí se describen aspectos fundamentales de la misma como una descripción del ecosistema, el establecimiento de los principales casos de uso o la definición de la **Arquitectura y Marco de Referencia** junto con sus componentes. En los numerosos repositorios pertenecientes a la organización se alojan las diferentes librerías elaboradas para la creación de una **aplicación móvil** para Android y IOS que permite a los usuarios gestionar sus credenciales.

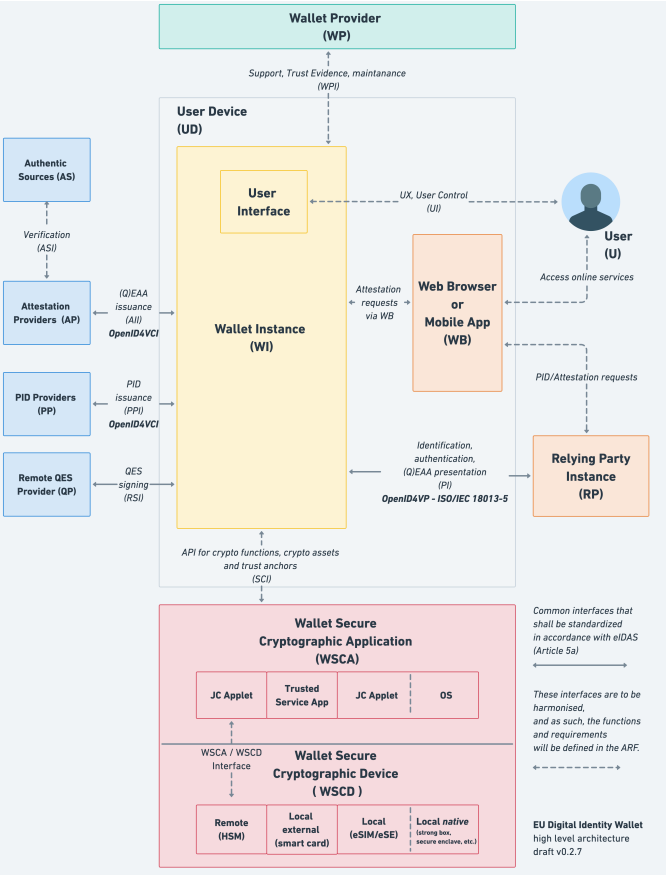

Figura 11: *Arquitectura de la EUDI Wallet.* Fuente [7]

3

Aplicabilidad de la Identidad Autosoberana

Analizando la situación actual descrita en el Estado del arte, resulta notable que tanto el paradigma en sí mismo como su adaptación al gran público europeo mediante la eIDAS traten de **reflejar** la información de entidades reales (entorno físico) a su equivalente digital.

De igual manera que se puede gestionar las credenciales, gracias al paradigma de la SSI el conjunto de dispositivos del Internet de las Cosas (IoT) es capaz de autenticarse y almacenar información referente a ellos mediante sus propios **certificados** (VC para objetos).

> **Observación 4**
>
> Cuando hablamos de dispositivos del IoT nos referimos a todo aquel que sea capaz de conectarse a la red Wi-Fi, que tenga una manera de identificarse inequívocamente en ella y que posea la funcionalidad de recoger información del entorno para su procesamiento. Ejemplos: Sensores, cámaras de vigilancia, teléfonos inteligentes, electrodomésticos ...

Por este motivo, la Red de Confianza Integrada (ITN) propone el uso de un nuevo término para referirse al DT de los dispositivos del IoT para así destacar el aspecto descentralizado del proceso de identificación, al que denominan Gemelo Digital Autosoberano (SSDT) [18]. Estos tendrán la capacidad de expedir, gestionar y consultar (dependiendo del actor) sus certificados mediante un único **registro** que haga las veces de VDR, mantenido por la propia ITN [9] siguiendo la misma idea propuesta por la EBSI para la implantación de la eIDAS.

Figura 12: *Actores y sus funciones en los SSDTs.* Fuente [9]

Como se ha podido comprobar, el paradigma de la SSI tiene la capacidad de ser aplicado en múltiples ámbitos igualmente innovadores con los que comparte fundamentos. De hecho, el alcance de esta situación es tal que estas aplicaciones, como el IoT ya comentado o los DTs que describiremos a continuación, suponen nuevos retos para el conjunto del sistema.

Por lo tanto, el presente estudio debe expandirse hasta abarcar el área de Gemelos Digitales para así conocer los Riesgos y amenazas intrínsecos al emplear el paradigma como alternativa a la centralización. Además, con el objetivo de completar el análisis teórico previo desde un punto de vista práctico, se elaborará una Prueba de concepto en la que se implementarán los componentes técnicos necesarios para reproducir el sistema en un entorno lo más real posible.

3.1. Gemelos Digitales.

El concepto de Gemelo Digital (DT) como lo conocemos actualmente tiene multitud de pequeñas variaciones según el contexto en el que se emplee y el aspecto a destacar del mismo, aunque no debe ser confundido con términos como Simulador, Avatar Digital o Huella Digital.

Algunos ejemplos de entornos digitalizables son el **industrial** mediante la replicación del proceso de producción, el **sanitario** con el modelado del estado de los pacientes o el **civil** para la creación de ciudades inteligentes tras monitorizar aspectos como el tráfico. Un caso en el que se puede ver hasta donde se puede extrapolar esta idea es en los esfuerzos por parte de NVIDIA en crear un DT de la Tierra para simular el efecto del cambio climático [4].

De tal manera, en el contexto de este escrito definiremos el concepto como "una representación virtual de un componente o sistema físico" [17], en el que se sobrentiende que tratamos de digitalizar la identidad de la persona mediante la creación de las VCs. En otras palabras, podemos decir que llegados a este punto del paradigma hemos derivado a un entorno de DTs ya que son igualmente distinguibles la propia persona **física** como su contraparte **digital**.

También es posible entenderlo de manera inversa, puesto que su capacidad de "caracterizar activos físicos a través de activos digitales" [2] hace resaltar la fuerte conexión que ambas partes han de mantener. Cualquier creación, actualización o eliminación de información en una de las partes debe verse reflejada en su equivalente para mantener siempre la concordancia.

3.1.1. Espacios de trabajo.

Siguiendo esta distinción, en el contexto de la SSI podemos encontrar:

- **Espacio Físico**, ámbito de los activos físicos.
  A él pertenecen aquellos mecanismos como los VDRs, las Carteras de Identidad Digital o los DIDs y sus derivados, cuya función consiste en modelar digitalmente los métodos tradicionales para la expedición de credenciales como DNIs, pasaportes, títulos educativos o cualquier otro tipo de documento acreditativo.

- **Espacio Digital**, ámbito de los activos digitales.
  Una vez nos desprendemos de la naturaleza real (soporte físisco) de la credencial mediante el uso de dichas herramientas, vemos que las VCs son capaces de realizar la misma función sin sustituir a su contraparte, puesto que ambos trabajan en espacios diferentes.

Determinados autores como [2] añaden a esta separación entre espacios un tercero al que denominan como "**Espacio de Comunicación**" que interconecta a ambos. Conformado por interfaces o servicios que manejasen los datos, en nuestro contexto se correspondería con el escenario conformado por los actores del paradigma, con sus relaciones y el uso que dan a sus dispositivos en los que guardan sus Carteras de Identidad Digital.

3.1.2. Arquitectura multicapa.

En [2] se realiza una completa labor de investigación que resulta en la definición de una arquitectura de un total de cuatro **capas de funcionalidad** para los sistemas de DTs:

- **Capa 1**, *difusión y adquisición de datos.*
  Es la capa más cercana al espacio físico y por ello interactúa directamente con los activos físicos. No solo tiene a cargo la recopilación de datos mediante **sensores** que envía a las capas superiores, sino que también posee **actuadores** con los que transmite la respuesta recibida para mantener la consistencia entre ambos activos.

- **Capa 2**, *gestión y sincronización de datos.*
  Previo al uso de estos datos heterogéneos (probablemente obtenidos de diversas fuentes) recibidos de la capa anterior, se debe realizar una serie de procesos de **normalización** y **enriquecimiento** que mejoren el rendimiento de los servicios de la tercera capa. Igualmente, se ha de controlar de forma **sincronizada** el flujo de información en ambos sentidos para lograr una correcta comunicación en la red.

- **Capa 3**, *modelado de datos y servicios adicionales.*
  La capa principal del la arquitectura es la encargada de, a través de modelos digitales creados a partir de los activos físicos y gracias a la información proporcionada, especificar los **estados** y **comportamientos** de estos para así poder analizar escenarios sin interferir en el mundo físico. Esto es posible mediante **servicios** que junto con la acción de los **sensores y actuadores virtuales** permiten la definición de los activos digitales.

- **Capa 4**, *visualización de datos y accesibilidad.*
  Como su nombre bien indica, facilita a los usuarios la consulta de los resultados obtenidos tras el modelado para la toma de decisiones. Otros mecanismos pueden ser definidos, como la gestión de la cadena de suministro o tener múltiples DTs en el mismo sistema.

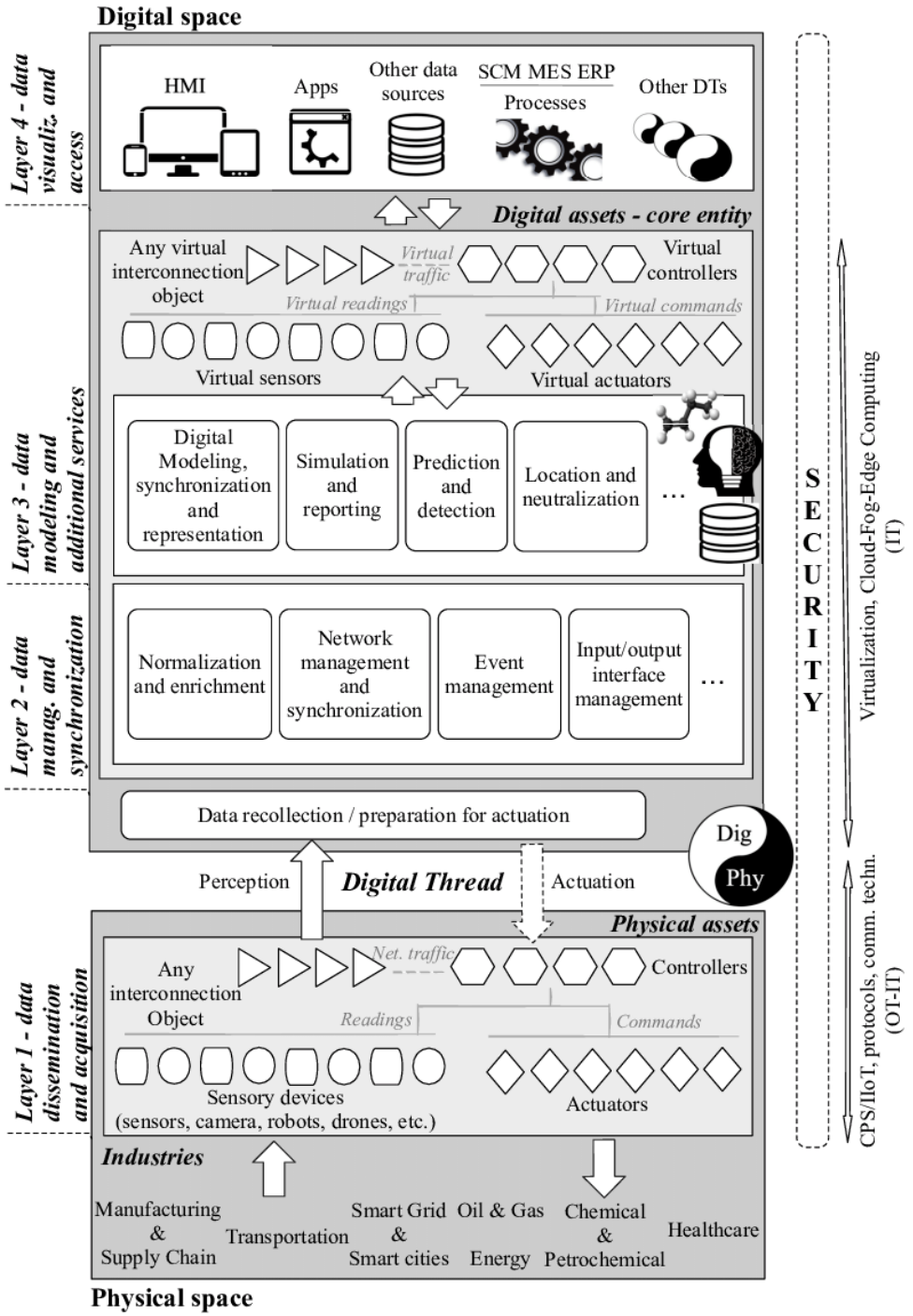

Figura 13: *Arquitectura multicapa para los DTs.* Fuente [2]

3.2. Riesgos y amenazas.

Una vez definida, es posible hacer uso de la arquitectura multicapa como base para **asociar** las vulnerabilidades propias de la SSI con las intrínsecas de DTs. Así pues, se realizará un análisis conjunto de su aplicabilidad para componentes y activos digitales. Las principales fuentes de información empleadas para la elaboración de esta sección son [2] para el ámbito de los DTs, además de [13] y [10] para el paradigma de la SSI.

3.2.1. Capa 1, difusión y adquisición de datos.

R1.1 Vulnerabilidades propias de la Blockchain.

Los VDRs garantizan la descentralización y transparencia al paradigma, pero son susceptibles a ataques a mecanismos propios como los Contratos inteligentes, ya sea explotando vulnerabilidades, desactivándolos, o cambiando las configuraciones.

Amenazas relacionadas en el ámbito de los DTs:

- **Escalado de privilegios [A1.1]:**
  Los atacantes sortearían la seguridad que la Blockchain proporciona si logran tomar el papel de administradores del VDR (o suplantar a otros actores) y así tener la capacidad de realizar acciones maliciosas de forma encubierta.
- **Denegación de servicio (DoS) [A1.2]:**
  El sistema es susceptible a ser saturado por peticiones benignas o maliciosas que en su conjunto supongan un problema para ser atendidas. Esto afectaría a la gestión del espacio físico y por tanto, a la creación de los activos digitales.

Amenazas mitigadas en el ámbito de los DTs:

- **Daño físico [A1.3]:**
  La principal cualidad de la Blockchain es la descentralización, puesto que su división en nodos permite la distribución de la información. Los protocolos de consenso garantizan que, si alguno de sus nodos ha sido vulnerado, los cambios propuestos por él no sean efectuados ya que el resto discreparía. No obstante, si el número de máquinas vulneradas es mayor al establecido (Quórum) se perdería esta barrera de seguridad y existiría riesgo de manipulación en el sistema.

R1.2 Suplantación de la identidad digital.

Aunque los mecanismos técnicos permiten afirmar de forma descentralizada que el portador de las VCs es quién dice ser, esto no garantiza que existan métodos para aprovechar esa asociación con fines maliciosos. De entre ellos, podemos destacar la obtención/eliminación de la clave privada o el aprovechamiento de los métodos de recuperación de la identidad.

Amenazas relacionadas en el ámbito de los DTs:

- **Extracción de información privada [A1.4]:**
  Si el dispositivo en el que se almacena la información sensible necesaria para la correcta identificación del usuario (como su clave privada o las propias VCs) resulta vulnerado, no existe ningún aspecto técnico que impida al atacante a usurpar su identidad. Esto puede evitarse mediante una correcta protección del dispositivo, como una buena política de contraseñas, usar la identificación biométrica o tener activado la Autenticación en dos pasos (2FA).

Amenazas mitigadas en el ámbito de los DTs:

- **Man-in-the-middle [A1.5]:**
  La identificación de los actores pertenecientes al espacio de comunicación es fundamental en este paradigma. Todos los mecanismos técnicos tienen como razón de ser alegar la identidad digital de los miembros mediante sus DIDs consultables, garantizando que el contacto sea directo entre las partes.

R1.3 Vulneración de la soberanía del dispositivo.

En la misma línea que la vulnerabilidad anterior, es de recalcar la vital importancia que el dispositivo tiene en el paradigma como almacén de las VCs. Aquel con acceso al mismo tiene el control de la identidad digital del usuario y a todos los efectos actúa como 'él' de manera reconocida y oficial en los entornos digitales que apliquen.

Por este motivo, cualquier acción maliciosa sobre su dispositivo es un ataque directo a su identidad digital, como por ejemplo el robo, la manipulación de los métodos de acceso o la inutilización física deliberada.

Amenazas relacionadas en el ámbito de los DTs:

- **Ataque al software [A1.6]:**
  El uso de software de terceros puede introducir vulnerabilidades al paradigma. Estas dependencias, incluidas como librerías o desarrolladas en el código para implementar algún componente (como la EUDI Wallet), también son propicias a ocasionar errores o generar alteraciones en el funcionamiento del sistema.
- **Dispositivos maliciosos [A1.7]:**
  Una vez comprometido, el dispositivo se convierte en la mejor manera para modificar el estado del DT, es decir, el atacante puede hacer y deshacer sin ningún impedimento en todo aspecto relativo a la identidad digital del usuario. Esto incluye la creación y eliminación de VCs, establecimiento de autorizados o el uso fraudulento de la misma en sitios o formas indeseadas.
- **Manipulación de hilos digitales [A1.8]:**
  Un caso concreto de la amenaza anterior es la introducción de código malicioso, configuraciones erróneas o datos falsos para producir una desincronización del espacio físico y afectar así al espacio digital, puesto que ambos mantienen una robusta relación. El eslabón débil es nuevamente la Cartera de Identidad Digital del usuario, la cual puede llegar a ser manipulada para hacer creer al usuario que el estado de su DT es diferente al real.

Amenazas mitigadas en el ámbito de los DTs:

- No aplican.

Análisis de la Capa 1

La Blockchain supone un gran avance para descentralizar la información necesaria para la identificación, pero no es capaz de solventar los problemas que ocasionaría el uso malintencionado (pero igualmente válido) de los mecanismos técnicos de la SSI.
Además, el dispositivo se convierte en pilar para la seguridad del sistema, haciendo más sencillo atacar el espacio físico que el digital (su protección queda en manos del usuario).

3.2.2. Capa 2, gestión y sincronización de datos.

R2.1 Manipulación sobre los nodos de la red.

Uno de los mayores retos para el paradigma de la SSI es la escalabilidad en su uso. La implantación masiva para el gran público supondrá un grave problema para la sincronización de los datos, por lo que soluciones basadas en la nube o software de virtualización para el aislamiento del sistema tomarán mayor relevancia.

Este hecho incluirá inevitablemente nuevas vulnerabilidades al paradigma, las cuales se añadirán a las ya presentes durante el proceso de establecimiento de los mismos. Ejemplo de ello es la propagación de mensajes falsificados para forzar el cierre, reinicio o aislamiento de los nodos, sin la necesidad de penetrar en el sistema.

Amenazas relacionadas en el ámbito de los DTs:

- **Ataque al software [A2.1]:**
  El despliegue de los nodos de la red en la nube o mediante software de virtualización puede ocasionar los mismos tipos de problemas mencionados en A1.6. Además de las dependencias de la tecnología empleada, se introducen otras propias del sistema operativo sobre el que se realiza el despliegue (Windows o Linux). En el caso de las aplicaciones móviles también habría que considerar Android e IOS, como es el caso de la EUDI Wallet.

- **Daño físico [A2.2]:**
  Los atacantes pueden comprometer el estado de las VCs tras tomar el control del sistema anfitrión, ya que podrían acceder a recursos sensibles, modificar configuraciones, parar servicios o ejecutar código. Sin embargo, los ataques a los servidores que conforman la nube resultan inusuales debido a la robusta capacidad de defensa que estos suelen poseen.

- **Escalado de privilegios [A2.3]:**
  Puede ser tentador para los atacantes realizar acciones maliciosas en el sistema que requieran privilegios elevados, como tomar el control del sistema anfitrión. Este hecho también aplica al software de virtualización, haciendo que el acceso directo sea una verdadera amenaza.

- **Nodos maliciosos [A2.4]:**

  De igual manera, si un nodo se ha visto comprometido puede actuar con fines maliciosos en la red realizando modificaciones del estado interno del mismo o denegando su operabilidad (reduciendo el número de nodos activos). Por los motivos descritos en A1.3, esto no supondría un riesgo para un número pequeño de nodos pero que podría ocasionar alteraciones graves en la red si aumenta.

Amenazas mitigadas en el ámbito de los DTs:

- **Pérdida de privacidad [A2.5]:**

  Esta amenaza hace referencia al uso indebido que las grandes corporaciones pueden llegar a hacer con los datos personales de sus usuarios, normalmente incorporando a su modelo de negocio la venta a terceros o el uso de técnicas computacionales para realizar análisis de mercado.

  El paradigma de la SSI, como alternativa descentralizada que es, no requiere del uso de estos servicios más allá de los motivos descritos en esta sección. Además, gracias a la transparencia que supone reflejar información 'sensible' en la Blockchain pública, la cantidad de datos privados se ve reducida.

R2.2 Ataque al espacio de comunicación.

Debido a que los grandes proveedores de servicios en la nube suelen ser demasiado seguros y como la solución que propone el paradigma de la SSI resulta ser lo suficientemente robusta, los principales vectores de ataque se concentran en los puntos de transmisión y recepción de datos. En este caso, el objetivo son los propios actores.

Amenazas relacionadas en el ámbito de los DTs:

- **Man-in-the-middle [A2.6]:**

  En toda comunicación llevada a cabo sobre una infraestructura de red existe la posibilidad de interceptación de los mensajes compartidos entre los actores. Esto puede ser evitado mediante el establecimiento de conexiones seguras, el empleo de mecanismos de cifrado para ocultar el contenido de los mensajes y el uso responsable del medio, evitando compartir información sensible.

- **Extracción de información privada [A2.7]:**
  Siguiendo la misma línea descrita en A2.3, si el usuario decide alojar en la nube su Cartera de Identidad Digital (la cual contiene sus VCs) o aislarla mediante un software de virtualización, existe la posibilidad que los administradores o usuarios privilegiados del sistema anfitrión realicen una labor de monitorización sobre sus comunicaciones e interceptar información sensible.
- **Denegación de servicio (DoS) [A2.8]:**
  Ante el impedimento de utilizar los servicios para acceder a los recursos necesarios para la identificación, independientemente de si han sido almacenados de manera local, en la nube o en una máquina virtual, el usuario no podría utilizarlos aún siendo el poseedor de las VCs allí alojadas. Por lo tanto, a efectos prácticos perder el acceso supone 'dejar de ser' digitalmente hablando.

Amenazas mitigadas en el ámbito de los DTs:

- No aplican.

Análisis de la Capa 2

El papel que realizan los componentes de esta capa seguramente sea el más complicado de lograr de manera segura, puesto que ninguno de los dos paradigmas solventan de manera efectiva la sincronización de los nodos de la red.

De hecho, uno de los puntos fuertes de la SSI es su robustez ante las acciones maliciosas que intentan 'romper' sus métodos criptográficos, pero esta resultaría sorteada si se obtiene acceso a toda la VC (o incluso al DID) para así usurpar la identidad.

Sucede algo parecido con la Autenticación en dos pasos (2FA), puesto que es más seguro si los códigos están asociados a un único dispositivo local sobre el que se tiene control absoluto. No obstante, sería correcto destacar que estos tienen un propósito diferente y carecen del fuerte aspecto interactivo que poseen las VCs.

3.2.3. Capa 3, modelado de datos y servicios adicionales.

R3.1 Uso fraudulento de las VCs.

A pesar de la fortaleza que los mecanismos técnicos de la SSI muestran, resultan ser insuficientes para evitar la manipulación o el uso malicioso por parte de atacantes.

Amenazas relacionadas en el ámbito de los DTs:

- **Ataque al software [A3.1]:**
  Durante el proceso de creación, modificación y eliminación de las VCs es posible que se cometan errores que introduzcan vulnerabilidades al sistema desde el espacio digital. Esta idea sigue la misma línea descrita en A1.6.
- **Extracción de información privada [A3.2]:**
  Tal y como se menciona en A2.7, si el atacante llega a tomar control de las VCs toda su potencial información privada que contengan puede ser extraída, incluyendo datos personales y referencias a credenciales físicas.
- **Pérdida de privacidad [A3.3]:**
  Vulnerando múltiples VCs de un mismo usuario, es posible correlacionar la información que contienen para inferir relaciones o asociaciones que puedan llegar a comprometer su privacidad.

Amenazas mitigadas en el ámbito de los DTs:

- **Manipulación de los datos contenidos [A3.4]:**
  Puesto que las VCs poseen mecanismos que preservan la integridad de los datos, la posibilidad de alterar su contenido para su posterior uso se ve invalidada.

Análisis de la Capa 3

Es destacable que la mejor manera para afectar al espacio digital sea mediante las capas anteriormente descritas. Una vez ha sido creada, la VC supone un punto de inflexión en la arquitectura del paradigma y pasan a ser un objetivo para los atacantes, los cuales tratarán de obtener toda información posible que esté contenida en ellas.

3.2.4. Capa 4, visualización de datos y accesibilidad.

R4.1 Distorsión de los datos en su representación.

En última instancia encontramos que, ante las protecciones que el sistema puede poseer para evitar la manipulación de los activos físicos (e incluso digitales), resulte tentador para los atacantes dirigir sus esfuerzos a comprometer la impresión del estado de las VCs para así aprovechar otras vulnerabilidades.

Amenazas relacionadas en el ámbito de los DTs:

- **Ataque al software [A4.1]:**
  Atacando directamente los componentes de la aplicación que muestra las VCs al usuario, se puede alterar el comportamiento de la misma sin la necesidad de haber afectado al funcionamiento de los mecanismos técnicos de la SSI.
- **Manipulación de la visualización de los datos [A4.2]:**
  Un caso particular de la amenaza anterior es, tras afectar a la correcta visualización de la información, originar en los usuarios ideas equivocadas que inciten a realizar acciones innecesarias. Para lograrlo, los atacantes pueden mostrar los datos de manera errónea e inconsistente, esconder determinada información o afectar la integridad de la misma.

Amenazas mitigadas en el ámbito de los DTs:

- **Aplicaciones maliciosas [A4.3]:**
  Gracias a soluciones de código abierto como la EUDI Wallet, el notable riesgo que supondría tener múltiples métodos de gestión de las VCs se ve reducido. Esto es no solo por establecer una única aplicación para su uso generalizado, sino por mantener la descentralización al almacenarlas en los dispositivos.

Análisis de la Capa 4

El principal riesgo asociado a esta capa es el engaño por parte de la aplicación al usuario. Además, comprobamos cómo la confianza puesta sobre la correcta implementación de la aplicación es máxima, por lo que toda transparencia posible resulta agradecida.

3.2.5. Balance general y tabla resumen.

Tras este intento por adecuar el paradigma de la SSI en un entorno de DTs, en el que se ha fusionado las capas de ambos ámbitos para analizar sus riesgos y amenazas, recopilamos las conclusiones obtenidas a lo largo del mismo para destacar que:

- Las principales vulnerabilidades surgen entre la **capa 1 y 2**.
- Están relacionadas con la **implementación** a nivel de software y comunicaciones.
- Los mecanismos técnicos de la SSI mitigan **algunas** de estas amenazas.
- Existen grandes **similitudes** entre ambos paradigmas en estos aspectos.

A continuación se detalla la tabla resumen que recoge el análisis realizado:

	Aspecto vulnerado	**Riesgo SSI intrínseco**	**Amenazas DT relacionadas**	**Amenazas DT mitigadas**
Capa 1	VDR	R1.1	A1.1, A1.2	A1.3
	Cartera de Identidad Digital	R1.2 R1.3	A1.4 A1.6, A1.7, A1.8	A1.5 No aplican
Capa 2	Nodo de la red	R2.1	A2.1, A2.2, A2.3, A2.4	A2.5
	Comunicaciones	R2.2	A2.6, A2.7, A2.8	No aplican
Capa 3	VC	R3.1	A3.1, A3.2, A3.3	A3.4
Capa 4	Representación	R4.1	A4.1, A4.2	A4.3

Cuadro 1: Asociación de los riesgos y amenazas en el conjunto de ambos paradigmas.

3.3. Prueba de concepto.

Con esta subsección se pretenden aplicar los contenidos expuestos en apartados descritos en esta misma sección desde un punto de vista más técnico y práctico. Para ello, se ha implementado un sistema capaz de reproducir una situación lo más parecida posible a la realidad.

En primer lugar se expondrán los flujos de acción disponibles mediante sus correspondientes diagramas, seguido de una visión general del comportamiento y aspecto del mismo, concluyendo finalmente con un breve análisis sobre las tecnologías empleadas.

3.3.1. Definición de los flujos de acción.

En el sistema intervendrán los propios actores del paradigma introducidos con anterioridad (Emisor, Titular y Verificador), a los cuales se añadirán los siguientes:

- **Registro**.
  El Titular debe adquirir un token de control de acceso al Servicio que es expedido por este actor. Este comportamiento es típico de protocolos tipo OpenID y gracias a la SSI se obtiene la garantía de identificar inequívocamente al usuario.

- **Servicio**.
  Representa el objetivo final del Titular, por el cual ha requerido de sus credenciales para identificarse. Dependiendo del flujo escogido variará el método de acceso al mismo.

Con el objetivo de facilitar la monitorización del sistema en su conjunto, se introducirá un nuevo cuasi-actor al que se denominará **main**. Como su propio nombre indica, será el ejecutor de la batería de pruebas (de ahora en adelante llamados **pasos**) y no es considerado como parte del sistema ya que no realiza ninguna función más allá de recibir y mostrar al usuario los mensajes obtenidos por el Titular.

A. El Titular debe pasar por el Registro para poder hacer uso del Servicio.

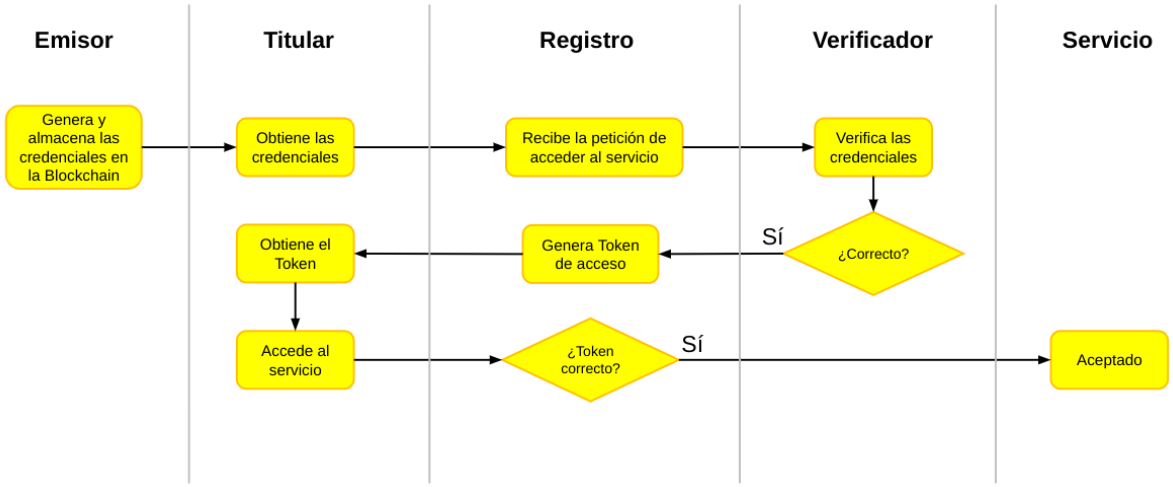

Figura 14: *Flujo de comunicaciones para la Prueba de Concepto A.*

B. El Titular puede interactuar con el Servicio, pero solo podrá valerse de él una vez se compruebe el token proporcionado en el Registro.

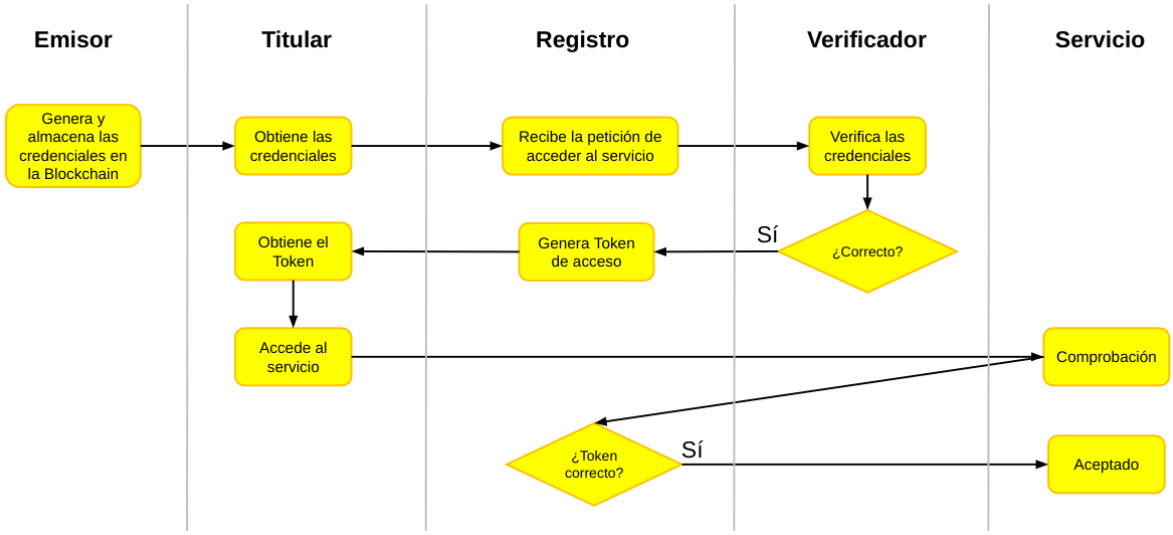

Figura 15: *Flujo de comunicaciones para la Prueba de Concepto B.*

En términos generales, se respetará el transcurso natural de las operaciones de los actores:

Paso 1. Actores involucrados: *Emisor*, *Titular*. El Titular pide al Emisor generar una VC para poder ser identificado y guarde su Hash en la Blockchain.

Paso 2. Actores involucrados: *Titular*, *Registro*, *Verificador*. El Titular realiza una petición de acceso al Registro, el cual genera un token tras la confirmación del Verificador.

Paso 3. Actores involucrados: *Titular*, *Registro*, *Servicio*. El Titular utiliza su token para autenticarse en el Registro y acceder al Servicio. También puede utilizarlo directamente en el Servicio, siendo necesaria una comprobación posterior en el Registro.

3.3.2. Visión general del sistema.

A continuación, se mostrará a grandes rasgos el funcionamiento del sistema desarrollado el cual se describirá con mayor grado de detalles técnicos en el apéndice de este mismo TFG. El primer aspecto que el usuario tendrá que decidir es cuál de los flujos descritos con anterioridad seguirá, teniendo que pinchar en uno de los dos botones azules designados para ello.

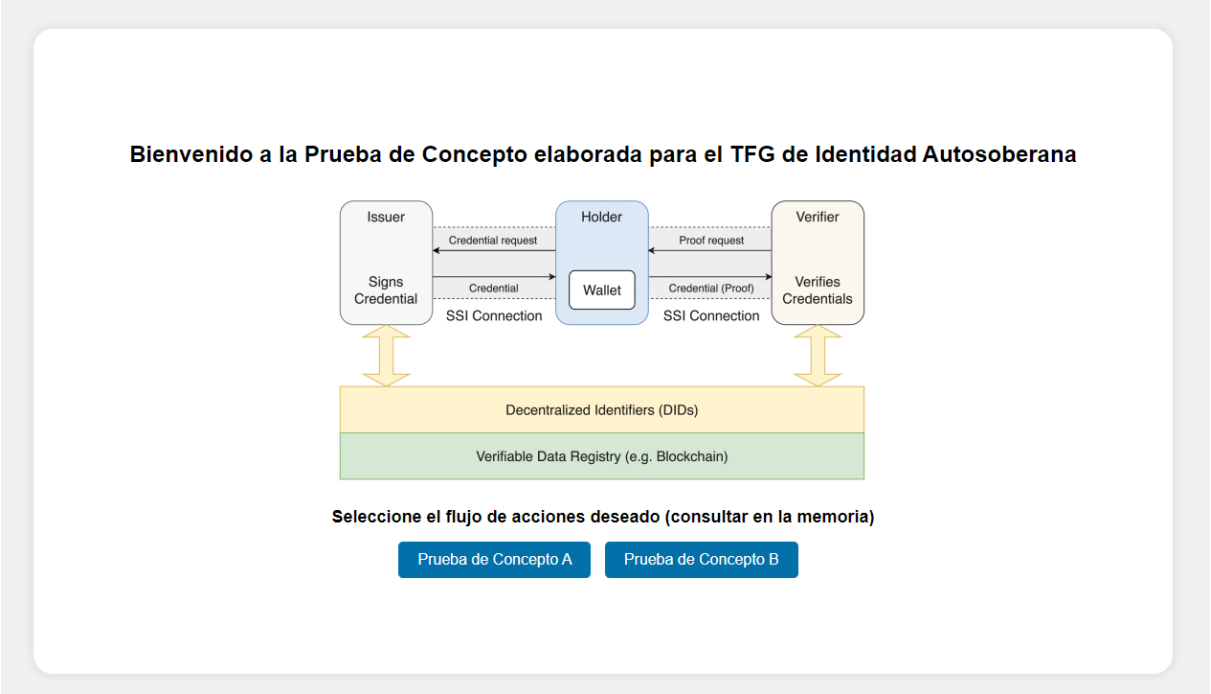

Figura 16: *Página principal del sistema.*

Dependiendo de su elección, dispondrá de las siguientes páginas las cuales incluyen:

- Un recuadro gris en el que se mostrará el estado final de las comunicaciones.
- Sendos botones azules correspondientes a cada paso del flujo en cuestión.
- Un botón rojo para volver a la página principal de inicio y escoger otro flujo.

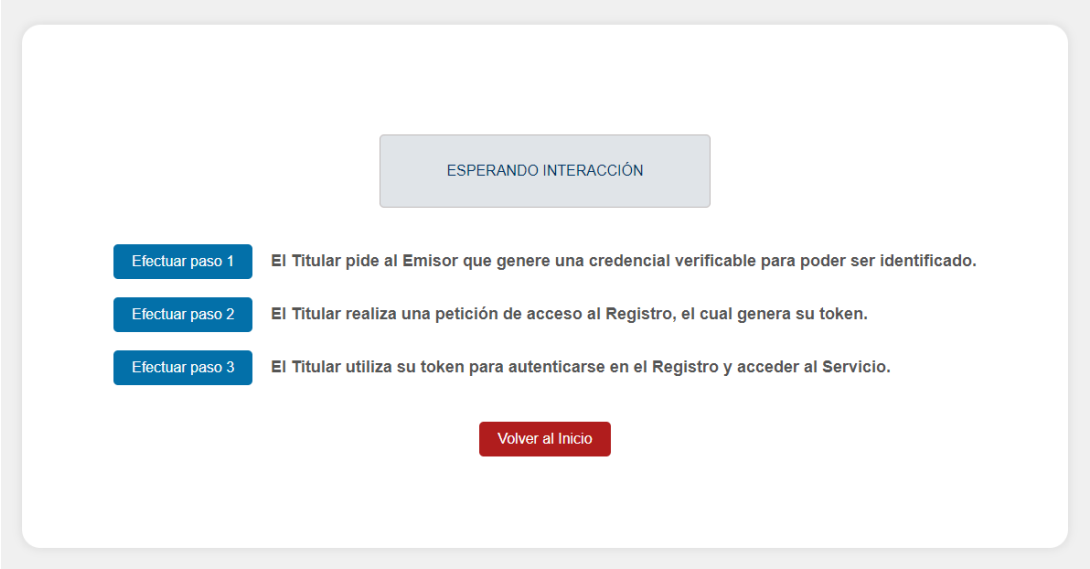

Figura 17: *Página de la Prueba de Concepto A.*

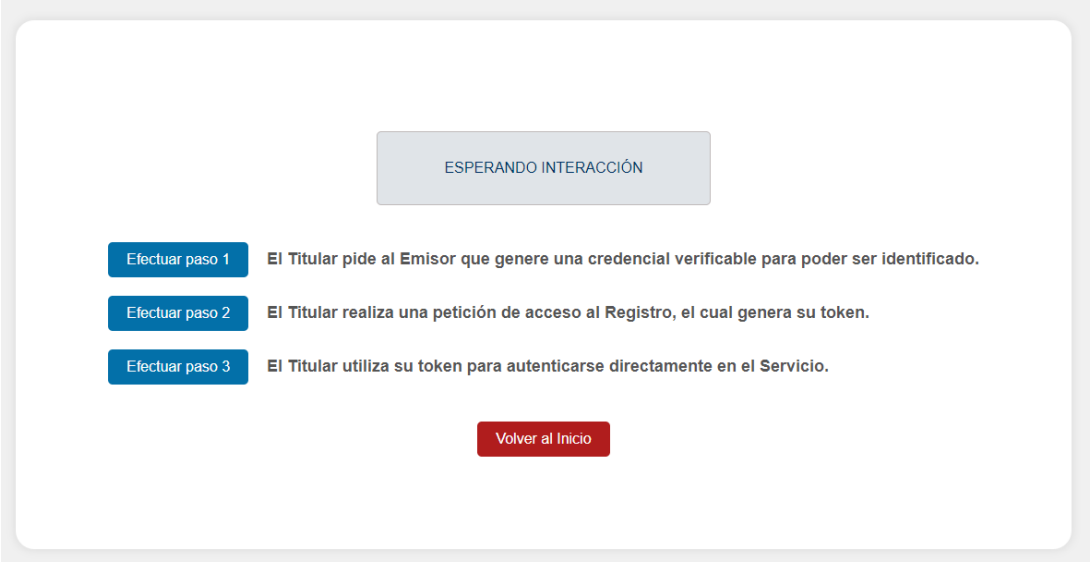

Figura 18: *Página de la Prueba de Concepto B.*

3.3.3. Análisis sobre las tecnologías empleadas.

Finalmente, se realizará una breve introducción a los medios técnicos que fueron empleados para la implementación de la prueba de concepto descrita con anterioridad. Con esto no se pretende dar una visión general de la tecnología, más bien destacar los motivos por los que fueron escogidos y la función que desempeñan en su conjunto.

- **Python**.

  La lógica interna del sistema estará implementada principalmente en *Python* y se basará en frameworks propios como *Flask* para establecer las conexiones necesarias durante las comunicaciones entre los componentes (actores) del mismo.

  La decisión de emplear *Python* se fundamenta en la simplicidad de su sintaxis y la consecuente legibilidad del código elaborado, además de la posibilidad de importar numerosas librerías relacionadas con el paradigma. Dichos aspectos, entre otros, han convertido a *Python* en uno de los lenguajes de programación más utilizados de los últimos tiempos.

  En el sistema, se conectarán los componentes con el intercambio de mensajes estructurados en formato JSON incluidos en peticiones HTTP, empleando para ello la librería Urllib (aunque pueden emplearse alternativas). Además, *Flask* permite la integración de recursos propios del desarrollo web, como HTML, CSS y JavaScript, los cuales servirán para acceder al monitorizador del sistema mediante el navegador.

Figura 19: *Logo oficial de* ***Python***.

- **Docker**.

  Uno de los requisitos indispensables del sistema es mantener los componentes aislados entre sí, ya que es necesario simular un entorno real en el que cada actor puede realizar su función en el paradigma de manera autónoma e independiente.

  Esto se conseguirá 'dockerizando' individualmente cada fichero que abstrae el comportamiento de los diferentes actores, generando así un conjunto de 'imágenes' que sean fáciles de 'desplegar' mediante su agrupación en un 'contenedor'. Además, gracias a este proceso se elimina la necesidad de tener instaladas en el dispositivo las dependencias necesarias para su correcto funcionamiento, agilizando entonces su portabilidad.

Figura 20: *Logo oficial de* ***Docker****.*

- **Kubernetes**.

  Aunque no sería necesario per se, es posible realizar un despliegue en *Kubernetes* para la mejor orquestación en 'pods' (conjunto de imágenes que conforman el contenedor) de nuestro sistema. Esta alternativa nos permite tener un mayor control sobre la gestión, escalado y exposición de los mismos, aunque suele estar pensada para entornos más complejos y de mayor envergadura que el nuestro. Cabe destacar que *Kubernetes* no es un sustituto de *Docker*, más bien es una extensión que amplía el alcance de las capacidades de *Docker*, empleando como base los recursos generados por él.

Figura 21: *Logo oficial de* ***Kubernetes****.*

- **PyCharm**.

  El Entorno de desarrollo empleado para estructurar el sistema es *PyCharm*, por un mero motivo de preferencia personal y al estar orientado a desarrollar código *Python*. Una facilidad que ofrece ante alternativas como *Visual Studio Code* es la integración nativa de herramientas de orquestación y gestión de entornos virtuales como *Kubernetes*.

Figura 22: *Logo oficial de* ***PyCharm***.

- **Besu Hyperledger**.

  En el apartado técnico de la Blockchain, se obtendrá desde Docker Hub la imagen del cliente de Ethereum, implementado en JAVA y de código abierto *Besu Hyperledger*.

  La característica clave por la que se ha escogido esta solución entre otras similares es lo adecuada que resulta para el sistema, ya que crea una red privada dentro un contenedor y permite configurar aspectos como el establecimiento de las comunicaciones con HTTP.

  Los actores interactuarán con la red mediante la libreria Web3 para realizar las pertinentes transacciones acordes a las necesidades de la SSI, disponiendo de funciones tras el despliegue del contrato elaborado en *Solidity* para ello. Ejemplo de esto es el almacenamiento de los DIDs y el Hash de las VC creados mediante la librería DIDKit de *Python*.

Figura 23: *Logo oficial de* ***Besu Hyperledger***.

4

Conclusiones

4.1. Reflexiones personales.

Debido al objetivo de este TFG, resultaría apropiado diferenciar las impresiones según el contexto en el que se han obtenido. Por lo tanto, hay que distinguir entre reflexiones de los mecanismos técnicos del paradigma y de su plan para la adopción.

4.1.1. Sobre el paradigma de la Identidad Autosoberana.

Gracias al estudio realizado, el conocimiento de causa adquirido permite desarrollar una opinión propia ante esta tecnología ya que tener preconcebida una idea sobre la misma suele ser un craso error, especialmente si somos escépticos ante la digitalización de aspectos críticos y sensibles como es la identidad propia.

1. Solución como alternativa a la centralización.

   Pensar que este paradigma es perfecto y no existen los inconvenientes es una simpleza. De hecho, seguramente sea más sencillo y seguro si mantenemos el modelo centralizado tradicional para el almacenamiento de las credenciales, pero nunca arreglaríamos la gran dependencia a determinados proveedores de servicios que tenemos.

   Si hay algo que ha demostrado el paradigma es la fortaleza y robustez técnica suficiente para ser una alternativa viable con la que poder identificarnos digitalmente de manera descentralizada e independiente (que no es poco). No obstante tiene aspectos en los que mejorar, como solventar los problemas de escalabilidad o de recuperación ante pérdida de las credenciales, tal y como hemos tratado en el escrito.

2. Críticas al diseño de las funciones de los actores.

   Tal y como se comentó en la *Observación 1*, el Titular no puede formar parte del sistema si no es mediante la interacción del Emisor con el VDR. Desde mi punto de vista, este hecho cuya intención es ofrecer protección ante potenciales interacciones maliciosas, limita la autonomía y libertad del futuro poseedor de las VCs en el paradigma.

   Pongamos el caso en el que Emisor considera oportuno no atender la petición del Titular (por cualquier motivo, ya sea legal o por intereses personales). Diríamos entonces que el Titular se encuentra 'vetado' de participar en el sistema aunque él quisiera ser parte.

4.1.2. Sobre el establecimiento de la Identidad Digital Europea.

En mi opinión, no considero que los mecanismos técnicos del paradigma conlleven algún tipo de retroceso en los derechos o libertades del usuario (más bien es todo lo contrario). Sin embargo, el balance general hace visible que algunas de las principales vulnerabilidades surgen tras una incorrecta implementación de algún componente. Entonces me pregunto, ¿podemos fiarnos como ciudadanos de esta solución para gestionar nuestra identidad digital?

1. Dificultades legales para su implantación.

   Actualmente existen dos protecciones (o escollos, según se mire) ante la implantación del paradigma para los ciudadanos europeos. Estos son el denominado como 'Derecho al olvido' y el Reglamento General de Protección de Datos (GDPR).

   El primero hace referencia a la eliminación de determinada información personal en los medios digitales. No es de extrañar que el uso de Blockchain en los VDRs pueda ir en contra de este principio por la propia naturaleza inmutable de dicha tecnología.

   El segundo choca directamente con la asociación de componentes técnicos como los DIDs y las VCs con la persona (o entidad) física a la que representan. Aunque en un principio no pareciera ser un problema, ya que esa información la posee el Titular de las credenciales, sí lo es ante la pérdida o uso malicioso de estas.

2. Europa como líder regulador en tecnología.

   En los últimos años, la Comisión Europea ha destacado por llevar la iniciativa global en la regulación de nuevas tecnologías (Inteligencia Artificial) y la innovación en los métodos tradicionales de pago (Euro Digital) e identificación digital (eIDAS). Personalmente, no descarto que se extrapolen estas medidas para abarcar el IoT tras realizar un balance sobre los resultados obtenidos con estas regulaciones, debido a que a efectos prácticos la gestión de certificados del ITN es idéntica que el propuesto por la EBSI para las VCs.

   Aunque esta decisión parece ser correcta, algunas voces destacan que puede llegar a perjudicar el correcto desarrollo de estas tecnologías. Además, no debe olvidarse que estas alternativas no deben bajo ningún concepto ser sustituto de los métodos físicos.

Conclusión General

Considero que es una mera cuestión de tiempo que adoptemos de forma voluntaria esta tecnología de la misma manera que ya lo hacemos con otras, como las transacciones bancarias inmediatas. Para ello, Europa ejerce una excelente labor de tutelaje y protección durante la implantación del paradigma de la SSI, ya sea mediante sus recursos legislativos o financieros. No obstante puede tornarse en una filtro innecesario el cual puede condicionar indirectamente numerosos aspectos de nuestro día a día.

4.2. Líneas futuras.

En lo referente a trabajos continuistas a la labor desarrollada en este TFG, existen determinados puntos que con gran probabilidad necesitarán actualizaciones recurrentes a medida que evolucionen los mecanismos técnicos del paradigma y se implanten al gran público.

Algunas ideas para futuros autores que mantengan la línea del estudio realizado son:

- Desarrollar en profundidad las **implementaciones** existentes del paradigma.
- Expandir el análisis de **riesgos y amenazas** con situaciones no contempladas.
- Incidir en otros **ámbitos** propensos a ser aplicados al paradigma, como el IoT.
- Realizar un **seguimiento** al plan de implantación de la eIDAS a los ciudadanos europeos.

Referencias

[1] *A brief history of web & authentication*. URL: https://medium.com/@tushar.vatsa/a-brief-history-of-web-authentication-8c1886d916fe.

[2] Cristina Alcaraz y Javier Lopez. «Digital Twin: A Comprehensive Survey of Security Threats». En: *IEEE Communications Surveys & Tutorials* 24.3 (2022), págs. 1475-1503. DOI: 10.1109/COMST.2022.3171465.

[3] *Decentralized Identifiers (DIDs) v1.0*. 2022. URL: https://www.w3.org/TR/did-core/.

[4] *Earth-2 Platform for Climate Change Modeling*. 2024. URL: https://www.nvidia.com/en-us/high-performance-computing/earth-2/.

[5] *EBSI Verifiable Credentials Framework*. URL: https://ec.europa.eu/digital-building-blocks/sites/display/EBSI/EBSI+Verifiable+Credentials.

[6] *Entry into Force of the Digital Identity Regulation*. 2024. URL: https://digital-strategy.ec.europa.eu/en/news/entry-force-digital-identity-regulation.

[7] *EU Digital Identity Wallet Architecture and Reference Framework*. URL: https://github.com/eu-digital-identity-wallet/eudi-doc-architecture-and-reference-framework/blob/main/docs/arf.md.

[8] *European Digital Identity*. URL: https://commission.europa.eu/strategy-and-policy/priorities-2019-2024/europe-fit-digital-age/european-digital-identity_es.

[9] *Finding Your Thing On the Blockchain: IoT in the Age of Standardized Global Identifiers*. 2023. URL: https://itntrust.medium.com/finding-your-thing-on-the-blockchain-iot-in-the-age-of-standardized-global-identifiers-2b76f53462ca.

[10] A. Grüner, A. Mühle, N. Lockenvitz et al. «Analyzing and comparing the security of self-sovereign identity management systems through threat modeling». En: *International Journal of Information Security* 22 (2023), págs. 1231-1248.

[11] *Identidad Auto Soberana (SSI)*. 2024. URL: https://extrimian.io/es/wikis/identidad-auto-soberana-ssi/.

[12] A. Satybaldy, M. Nowostawski y J. Ellingsen. «Self-sovereign identity systems: Evaluation framework». En: *IFIP Advances in Information and Communication Technology*. Vol. 576. 2020, págs. 447-461. DOI: 10.1007/978-3-030-42504-3_28.

[13] Abylay Satybaldy, Md. Sadek Ferdous y Mariusz Nowostawski. «A Taxonomy of Challenges for Self-Sovereign Identity Systems». En: *IEEE Access* 12 (2024), págs. 16151-16177. DOI: 10.1109/ACCESS.2024.3357940.

[14] *SSI eIDAS Bridge*. URL: https://joinup.ec.europa.eu/collection/ssi-eidas-bridge/technical-deliverables.

[15] *Verifiable Credentials Data Model v2.0*. 2024. URL: https://www.w3.org/TR/vc-data-model-2.0/.

[16] *W3C VCs and VPs on EBSI*. URL: https://hub.ebsi.eu/vc-framework/w3c-vc-vp.

[17] *What Is a Digital Twin?* 2021. URL: https://blogs.nvidia.com/blog/what-is-a-digital-twin/.

[18] *When Things Know What They Are: IoT in the Age of Self-Sovereign Identity*. 2022. URL: https://itntrust.medium.com/when-things-know-what-they-are-iot-in-the-age-of-self-sovereign-identity-7446982489b9.

Glosario

Autenticación en dos pasos (2FA) Verificación realizada tras la tradicional comprobación del nombre de usuario y la contraseña asociada a la cuenta con la que se quiere iniciar sesión. Con ella, se requiere de una acción adicional desde el dispositivo configurado (como introducir un código autogenerado). 31, 35

Avatar Digital Representación visual del usuario en un determinado entorno digital. 26

contratos inteligentes Programa informático almacenado en la Blockchain que es capaz de efectuar acciones (ejecutarse) cuando se den las condiciones necesarias. 30

criptodivisas Método de intercambio de monedas digitales que emplea criptografía para realizar intercambios seguros y estructuras de datos como Blockchain para garantizar la descentralización. 16

entorno de desarrollo Conjunto de herramientas técnicas integradas en una misma de interfaz en la que cada una de ellas cumple una función determinada durante la creación de recursos informáticos, principalmente software. El principal cometido de este es facilitar la labor del programador. 45

firma digital Mecanismo criptográfico utilizado para garantizar la autenticidad e integridad de un mensaje, documento o cualquier otro tipo de dato digital. 20

Github Principal plataforma de alojamiento de proyectos tecnológicos, en los que cualquier persona puede visualizar el código fuente de los mismos. 23

hash Función criptográfica capaz de transformar una entrada de datos de cualquier tamaño en otra de longitud fija, la cual es única para la entrada proporcionada (por lo que pequeños cambios generan una salida completamente distinta) e irreversible (no se puede obtener la entrada a partir de la salida). 12, 41, 45

HTTPS Protocolo construido sobre HTTP que introduce una capa de seguridad (SSL/TLS) que permite el intercambio de información entre dispositivos de manera segura. 12

Huella Digital Conjunto de consecuencias reflejadas en el entorno digital del usuario a raíz de sus acciones. Esta información suele quedar plasmada en forma de datos rastreables que permiten reconstruir el modo de proceder del usuario. 26

Identificador Uniforme de Recursos Un URI es el formato de identificador estándar para todos los recursos en la World Wide Web. Más información en [3]. 17

OpenID Este estándar permite a los usuarios identificarse en los diferentes sitios web con una única cuenta para así evitar la creación de multitud de ellas. Un claro ejemplo de ello es el inicio de sesión mediante una cuenta de Google. 39

Simulador Especializado en el entorno digital de un sistema, trata de simplificar la complejidad intrínseca del entorno físico obviando los modelos con los que se representarían las características que requerirían una constante comunicación entre ambos [2]. 26

Acrónimos

DID Identificador Descentralizado. 12, 14, 17–19, 21, 27, 31, 35, 45, 48

DT Gemelo Digital. 2, 7, 25–38

EBSI Infraestructura Europea de Servicios Blockchain. 22, 25, 49

eIDAS Identidad Digital Europea. 2, 6, 7, 21, 25, 48, 49

ESSIF Marco Europeo de Identidad Soberana. 22

EUDI Wallet Cartera de Identidad Digital Europea. 23, 32, 33, 37

GDPR Reglamento General de Protección de Datos. 22, 48

IoT Internet de las Cosas. 2, 8, 25, 26, 49

ITN Red de Confianza Integrada. 25, 49

SSDT Gemelo Digital Autosoberano. 25, 26

SSI Identidad Autosoberana. 2, 6–8, 13–16, 21, 25–27, 30, 32–39, 45, 47, 49, 58

TFG Trabajo Fin de Grado. 2, 15, 41, 47, 49

VC Credencial Verificable. 12, 19–21, 25, 27, 31–33, 35–38, 41, 45, 48, 49, 60, 62

VDR Registro de Datos Verificables. 18, 19, 22, 25, 27, 30, 38, 48

W3C World Wide Web Consortium. 16

Apéndice A
Configuración de la prueba de concepto.

A.1. Manual de usuario.

La prueba de concepto requiere tener instalada la distribución de **Docker** adecuada al sistema operativo del dispositivo en el que se quiere ejecutar. De esta manera, únicamente se necesita que el usuario introduzca en una terminal abierta desde el directorio raíz del proyecto (donde se encuentra el archivo docker-compose.yml) el siguiente comando:

```
docker compose up --build
```

Tras este paso, se generará la siguiente estructura de contenedores que se mantendrán activos aunque el usuario cierre la terminal y permitirán el correcto funcionamiento del sistema:

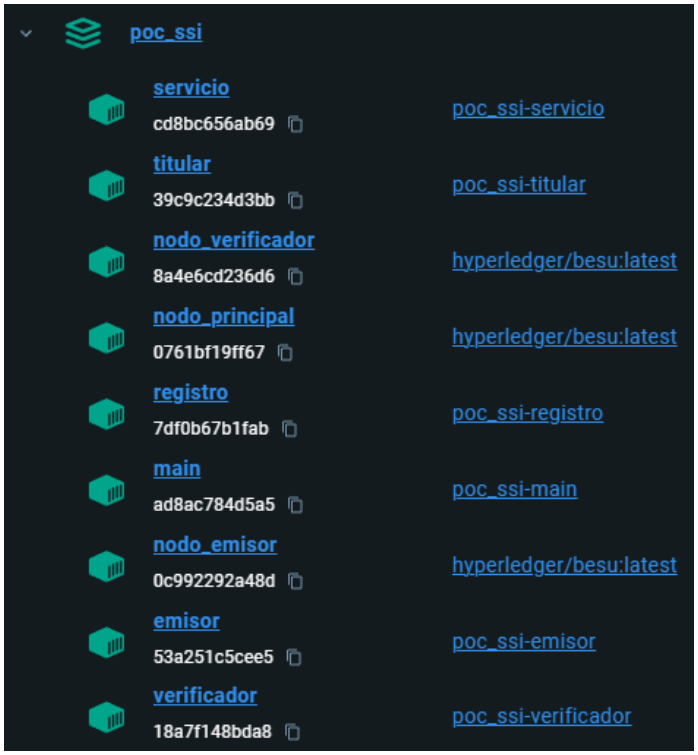

Figura 24: *Visualización de los contenedores en Docker Desktop.*

Acudiendo a la dirección **http://127.0.0.1:5000** desde cualquier navegador, el usuario podrá seguir el transcurso descrito en la sección de la prueba de concepto. También podrá acceder:

1. Al contenido de las variables del Emisor, mediante la url: **http://127.0.0.1:5001**
2. Al contenido de las variables del Titular, mediante la url: **http://127.0.0.1:5002**
3. Al contenido de las variables del Verificador, mediante la url: **http://127.0.0.1:5003**
4. Al contenido de las variables del Registro, mediante la url: **http://127.0.0.1:5004**
5. Al contenido de las variables del Servicio, mediante la url: **http://127.0.0.1:5005**

Observación

Se recomienda refrescar las páginas de los actores involucrados cada vez que se efectúe un paso del flujo elegido. De esta manera, se podrá conocer el estado actual de cada componente de la SSI y seguir su evolución durante la prueba de concepto.

Una vez logrado el objetivo por el que se quiso ejecutar el sistema, se deberá terminar adecuadamente con la actividad de los contenedores. Esto se puede realizar de dos maneras:

- Si la terminal sigue estando abierta, es suficiente con seleccionarla y pulsar las teclas CONTROL + C para así mandar una señal que terminará con los procesos en ejecución.
- Si por el contrario la hemos cerrado, hemos de realizar el siguiente comando:

```
docker stop $(docker ps -q)
```

Finalmente, resultaría correcto eliminar por completo todo elemento creado por el proyecto. Tras la ejecución de este comando desde cualquier terminal, se eliminará todo contenedor, volumen de datos e imagen creado por el sistema en el dispositivo del usuario.

```
docker compose -p poc_ssi down --rmi all -v
```

Alternativamente, puede realizarse este mismo proceso mediante la interfaz de usuario de Docker Desktop en caso de haberse instalado previamente (recomendado por su simplicidad).

A.2. Características de diseño.

Como puede haberse comprobado, el principal aspecto a destacar del sistema elaborado es su cómoda accesibilidad desde el navegador. Esto no fue considerado en primer instancia, ya que la visualización en terminal de las comunicaciones efectuadas y la posibilidad de acceder al propio contenedor parecían suficientes para monitorizar la prueba de concepto.

No obstante, el elevado número de actores y los frecuentes mensajes de los nodos que conforman la red Blockchain hacen demasiado engorrosa la visualización desde la terminal. Una solución a esta situación es acceder directamente a los registros (logs) de los contenedores para conocer su actividad individualmente, con el coste de perder la visión en conjunto del sistema.

Destacar que estas alternativas siguen estando disponibles aunque no son indispensables para realizar la función de monitorización, ya que gracias a determinadas tecnologías como *Flask* es posible acceder desde el navegador mientras se realizan comunicaciones en segundo plano.

```
nodo_principal    | 2024-09-10 12:11:38.781+00:00 | nioEventLoopGroup-3-3 | INFO  | FullSyncTargetManager | Unable to
 find sync target. Currently checking 2 peers for usefulness
nodo_principal    | 2024-09-10 12:11:43.785+00:00 | EthScheduler-Timer-0 | INFO  | FullSyncTargetManager | Unable to
find sync target. Currently checking 2 peers for usefulness
nodo_principal    | 2024-09-10 12:11:48.787+00:00 | EthScheduler-Timer-0 | INFO  | FullSyncTargetManager | Unable to
find sync target. Currently checking 2 peers for usefulness
nodo_principal    | 2024-09-10 12:11:51.036+00:00 | MinerExecutor | INFO  | BlockMiner | Produced #2 / 1 tx /
n=0ms, txsSelection=21ms, blockAssembled=3ms, importingBlock=5ms, notifyListeners=1ms, log=0ms)
nodo_emisor       | 2024-09-10 12:11:51.084+00:00 | EthScheduler-Workers-0 | INFO  | PersistBlockTask | Imported #2 /
 1 tx / 0 om / 22,980 (0.2%) gas / (0xa94deaf808fea21e828cd4a32622c1fb61c7dc98cbc18f4b62e73108552489f7) in 0.035s. Pe
ers: 2
nodo_verificador  | 2024-09-10 12:11:51.084+00:00 | EthScheduler-Workers-0 | INFO  | PersistBlockTask | Imported #2 /
 1 tx / 0 om / 22,980 (0.2%) gas / (0xa94deaf808fea21e828cd4a32622c1fb61c7dc98cbc18f4b62e73108552489f7) in 0.037s. Pe
ers: 2
emisor            | 172.20.0.6 - - [10/Sep/2024 12:11:51] "POST /generar_credencial HTTP/1.1" 200 -
titular           | 172.20.0.3 - - [10/Sep/2024 12:11:51] "GET /pedir_credencial HTTP/1.1" 200 -
main              | 172.20.0.1 - - [10/Sep/2024 12:11:51] "POST /poc_a_paso/1 HTTP/1.1" 200 -
verificador       | 172.20.0.7 - - [10/Sep/2024 12:11:51] "POST /comprobar_identidad HTTP/1.1" 200 -
registro          | 172.20.0.6 - - [10/Sep/2024 12:11:51] "POST /gestionar_acceso HTTP/1.1" 200 -
titular           | 172.20.0.3 - - [10/Sep/2024 12:11:51] "GET /obtener_acceso HTTP/1.1" 200 -
main              | 172.20.0.1 - - [10/Sep/2024 12:11:51] "POST /poc_a_paso/2 HTTP/1.1" 200 -
servicio          | 172.20.0.7 - - [10/Sep/2024 12:11:52] "POST /nuevo_actor HTTP/1.1" 200 -
registro          | 172.20.0.6 - - [10/Sep/2024 12:11:52] "POST /comprobar_token HTTP/1.1" 200 -
titular           | 172.20.0.3 - - [10/Sep/2024 12:11:52] "GET /acceder_servicio_mediante_registro HTTP/1.1" 200 -
main              | 172.20.0.1 - - [10/Sep/2024 12:11:52] "POST /poc_a_paso/3 HTTP/1.1" 200 -
```

Figura 25: *Ejemplo de comunicaciones entre contenedores durante los flujos.*

Estos mensajes son enviados y recibidos mediante los endpoints de *Flask* definidos en *Python*. Por ejemplo, el siguiente fragmento de código es utilizado por el Titular para obtener la VC creada por el Emisor, correspondiente al Paso 1 en cualquier flujo escogido (común a ambos).

```
@app.route('/pedir_credencial', methods=['GET'])  # Paso 1.
def pedir_credencial():
    # Construir la petición para el endpoint.
    datos_raw = {'did': titular.did}
    datos_bytes = json.dumps(datos_raw).encode('utf-8')
    peticion = titular.realizar_peticion(emisor_url + '/
   generar_credencial', datos_bytes, 'POST')

    try:
        # Obtener el contenido de la respuesta.
        respuesta = urllib.request.urlopen(peticion)
        credencial = str(respuesta.read().decode('utf-8'))

        # Realizar las acciones pertinentes.
        titular.guardar_credencial(credencial)

        # Comunicar la resolución definitiva.
        return jsonify({"Respuesta": "La credencial **SI** ha
   sido recibida y guardada."}), 200
    except urllib.error.HTTPError:
        return jsonify({"Respuesta": "La credencial **NO** ha
   sido recibida ni guardada."}), 400
```

En el sistema, todo endpoint comparte una estructura similar a la descrita ya que necesitan:

- Enviar algún tipo de información a otro/s actores para cumplimentar una petición.
- Recibir un mensaje con la respuesta obtenida distinguiendo los casos de éxito y fracaso.
- Comunicar el estado final de su petición a otro endpoint que haya llamado al actual. En el caso del Titular, es el cuasi-actor main el que requiere de confirmación.

Con el objetivo de definir una distribución de los archivos creados en la prueba de concepto previa a la contenedorización de los actores, se diseña la siguiente estructura de carpetas:

- En '**Actores**' se recogen un total de cinco subcarpetas asociadas a cada uno de los actores que participan en el sistema. En ellas se guardarán aquellos ficheros necesarios para la autonomía del mismo, como el Dockerfile o su *Python* correspondiente.
- En '**Blockchain**' se definen los recursos requeridos por *Besu* para el establecimiento de la red, como por ejemplo el genesis.json que describe su primer bloque o el propio contrato escrito en *Solidity*, además de numerosos archivos de configuración de los nodos.
- En '**Principal**' se encuentra el cuasi-actor main que supervisa el conjunto del sistema. Dentro de esta carpeta están vinculados los flujos a la batería principal de pruebas, la cual requiere de recursos web para visualizar la prueba de concepto en el navegador.

Además, en el directorio raíz del proyecto (PoC_SSI) se encuentra el docker-compose.yml del cual depende la correcta orquestación de los diferentes contenedores del sistema.

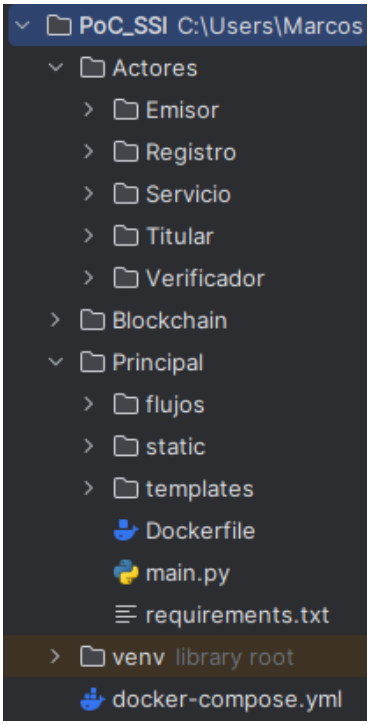

Figura 26: *Estructura de ficheros del proyecto.*

A.3. Limitaciones y aspectos a mejorar.

Aunque el proceso de elaboración de la prueba de concepto ha resultado satisfactorio, han surgido numerosas dificultades técnicas durante su implementación. Ya sea por la falta de documentación actualizada o por una excesiva complejidad intrínseca, se han originado algunos desajustes leves entre la idea original a desarrollar y el resultado final:

1. **Inicialización de la red Besu**.

   El sistema requiere de un notable tiempo para gestionar el primer paso, creación de VCs. Esto es debido a que debe desplegarse el contrato, crearse al menos un bloque vacío y que los validadores lo minen. Se origina por lo tanto, un aparente congelamiento del sistema del cual el usuario debe esperar y no pulsar ningún otro botón del flujo a seguir.

   En una primera instancia también se trató de establecer la red 'gas free', de manera que las transacciones no consumieran recursos. Tras el fallido intento, se optó por preestablecer cuentas con saldo positivo para ajustarse a un escenario lo más real posible.

2. **Volúmenes no definidos para los contenedores Docker**.

   Una vez se ejecuta el comando de arranque del sistema, las variables de los actores sufren inevitables modificaciones durante el trascurso del flujo. Sin embargo, estos cambios se pierden al parar los contenedores. Esta situación no ocurre con la Blockchain, teniendo asociada una serie de volúmenes que mantienen un registro de las transacciones.

3. **Despliegue del sistema en Kubernetes**.

   Como se ha podido comprobar, finalmente no se elaborado ningún tipo de recurso que facilite el despliegue mediante Kubernetes, principalmente por motivos de tiempo y por el tamaño del proyecto. No obstante, existen herramientas como *Kompose* que permiten transformar el archivo docker-compose.yml en recursos para *Kubernetes*, aunque se requerirá la modificación y ajuste de algunos de ellos para su correcto funcionamiento.

UNIVERSIDAD DE MÁLAGA | uma.es

E.T.S de Ingeniería Informática
Bulevar Louis Pasteur, 35
Campus de Teatinos
29071 Málaga

E.T.S. DE INGENIERÍA INFORMÁTICA