Sistema de verificación de edad para el acceso a contenidos en línea

Protocolo de verificación de edad

Versión 1

30 de junio de 2024

Plan de Recuperación, Transformación y Resiliencia
Financiado por la Unión Europea NextGenerationEU
España | digital 2026

AUTOR	Ministerio para la Transformación Digital y de la Función Pública
PROYECTO	Cartera Digital [BETA]
NOMBRE DEL DOCUMENTO	Protocolo de verificación de edad Sistema de verificación de edad para el acceso a contenidos en línea

Control de Versiones del Documento

VERSIÓN	AUTOR	FECHA	DESCRIPCIÓN
V1	Ministerio para la Transformación Digital y de la Función Pública	30-06-2024	Versión inicial

Índice

RELACIÓN DE FIGURAS

RELACIÓN DE TABLAS

1 INTRODUCCIÓN

Esta especificación técnica es parte de las funcionalidades del sistema de verificación de edad para el control en el acceso a contenidos para personas adultas de las personas menores de edad, que el Ministerio para la Transformación Digital y de la Función Pública (MTDFP) está definiendo e implementando.

La principal condición para esta especificación es que en ningún caso la persona que acredite la mayoría de edad ha de aportar información que permita su identificación o rastreo en Internet, y las plataformas de contenidos deben obtener, por tanto, la información mínima necesaria de los usuarios, siguiendo el principio de minimización de la divulgación de datos.

No es objeto de este documento cómo es el proceso de obtención de esa credencial, ni el proceso de consulta a una fuente auténtica previo a la generación de la citada credencial[1].

Este documento detalla **el protocolo de comunicación entre la aplicación móvil** del usuario final **y el proveedor** de contenidos **para la verificación de la mayoría de edad**. Incluye la **especificación técnica del proveedor de contenidos restringidos para adultos** para **garantizar la interoperabilidad**, en el ámbito nacional, entre las **plataformas con contenidos para adultos**.

Con este documento tanto el proveedor de contenidos para adultos como los desarrolladores de aplicaciones móviles deben poder implementar los flujos de información necesarios siguiendo los estándares y protocolos definidos con este fin.

Este documento supone que existe una aplicación móvil, Cartera Digital [BETA] será una de ella, en la cual un usuario ha podido almacenar de forma segura una credencial verificable que acredita que este es mayor de edad y, por tanto, **está autorizado a acceder a contenidos para adultos** (en ningún caso se comparte la edad). Mediante la credencial, el usuario final demuestra su mayoría de edad en el acceso a plataformas con contenidos restringido para adultos que tendrán la obligación de verificar este atributo.

> NOTA - Esta solución ha sido diseñada y planteada considerando el estado del arte actual en diferentes tecnologías criptográficas ampliamente extendidas y los principios que se están desarrollando en el reglamento eIDAS2[2]. En cualquier caso, no es una implementación completa de este reglamento que aún se está desarrollando.
>
> Se seguirá trabajando y mejorando en todos los procesos según vaya evolucionando el reglamento eIDAS2 y tecnologías de ZKP[3].

[1] El protocolo que detalla como Cartera Digital [BETA] realiza la emisión de las credenciales y su presentación se encuentra en el documento "Especificación de Uso Credencial Mayoría de Edad"

[2] Reglamento (UE) 2024/1183 del Parlamento Europeo y del Consejo, de 11 de abril de 2024, por el que se modifica el Reglamento (UE) nº 910/2014 en lo que respecta al establecimiento del marco europeo de identidad digital)

[3] Zero Knowledge Protocol, sistemas criptográficos altamente seguros focalizados en la revelación mínima de información

1.1 Alcance

Este documento contempla todos los elementos necesarios para que:

- Las **plataformas** que contengan contenidos para adultos puedan **solicitar, recibir y verificar credenciales de mayoría de edad**.
- Los proveedores de contenido para adultos puedan desarrollar la solución tanto en **plataformas web** como en **aplicaciones móviles**.
- Las **aplicaciones móviles que tengan la credencial** de **"mayoría de edad" pueden presentarla a las plataformas** de contenido para adultos.

1.2 Diccionario

- **Fuentes Auténticas de Información:** Fuentes de donde el emisor de la credencial extrae los atributos que posteriormente se incluyen en la credencial verificable.
- ***Decentralized Identifier* (DID):** Identificador descentralizado definido en [DID-Core]. Los DIDs de la presente solución serán generados mediante el método especificado en [DID-Key].
- **Solicitud de evidencia:** Petición realizada por un verificador en la que se indica al propietario de las credenciales, qué credenciales debe presentar a este y en qué formato debe de hacerlo.
- **Evidencia:** Respuesta a la solicitud de evidencia realizada por un verificador en la que se incluyen la credencial o credenciales solicitadas.
- **Listas blancas:** Se trata de una lista de entidades que son consideradas de confianza para realizar determinadas acciones, como puede ser, por ejemplo, solicitar cierto tipo de credenciales para su posterior verificación.
- ***Deep Link*:** Es una URL que lleva a un usuario directamente a una ubicación específica dentro de una aplicación o sitio web, en lugar de simplemente abrir la página de inicio.
- ***Universal Link*:** Es una tecnología introducida por Apple para iOS, que permite a los enlaces web abrir directamente el contenido relevante en una aplicación, si está instalada, o en el navegador web, si la aplicación no está instalada.
- ***App Link*:** Es el equivalente de *Universal Links* pero para la plataforma Android. Introducido por Google, los *App Links* permiten que las URLs web se abran directamente en una aplicación específica si está instalada, y en el navegador web si no lo está.

1.3 Actores

El siguiente diagrama muestra los actores del ecosistema:

- **Emisor de Credencial:** Solución encargada de generar la credencial de mayoría de edad. En primer lugar, extraerá los datos necesarios de las fuentes auténticas de información para generar una credencial de mayoría de edad que acredita que el titular de esta tiene más de dieciocho años (según la Constitución Española). Dicha

credencial incorpora la firma del emisor, de forma que terceros puedan validar quién es la entidad emisora.

- **Aplicación móvil Cartera Digital [BETA]:** aplicación móvil desarrollada por el Ministerio para la Transformación Digital y de la Función Pública que en el contexto de este documento contendrá el par de claves criptográficas, pública y privada, y la credencial de mayoría de edad.
- **Proveedor de contenidos (Verificador):** Componente encargado de solicitar la credencial de mayoría de edad al usuario final y realizar las verificaciones correspondientes para permitir o denegar el acceso al contenido restringido para adultos.

Figura 1. Componentes solución general

El diagrama muestra los protocolos de comunicación de la solución general como son el protocolo de emisión de credenciales mediante el cual la aplicación móvil obtiene la credencial de mayoría de edad y el protocolo de presentación de la credencial de mayoría de edad mediante el cual la aplicación móvil presenta la credencial al proveedor de contenidos para obtener acceso al contenido restringido para adultos. Dado que este documento tiene como objetivo la definición de la solución técnica para la solicitud y verificación de la credencial de mayoría de edad, se tratará únicamente el protocolo de presentación de la credencial de mayoría de edad que se realizará siguiendo la especificación **OpenID for Verifiable Presentations [OpenID4VP]**, así como en el modelo de los datos que comparten ambas partes durante dicha comunicación.

El marco de confianza está basado en listas blancas gestionadas por una autoridad certificadora. Desde el prisma del proveedor de contenidos, sólo es necesario tratar la lista de proveedores de contenido de confianza, donde deberán registrarse previamente y la lista de emisores de confianza, la cual deberá ser consultada para verificar que la credencial de mayoría de edad ha sido emitida por un emisor de confianza.

1.4 Flujo General

El siguiente diagrama representa el flujo de comunicación a alto nivel de los actores introducidos previamente, que componen la solución técnica común. Mediante el recuadro verde se resalta la solución técnica que se aborda en este documento.

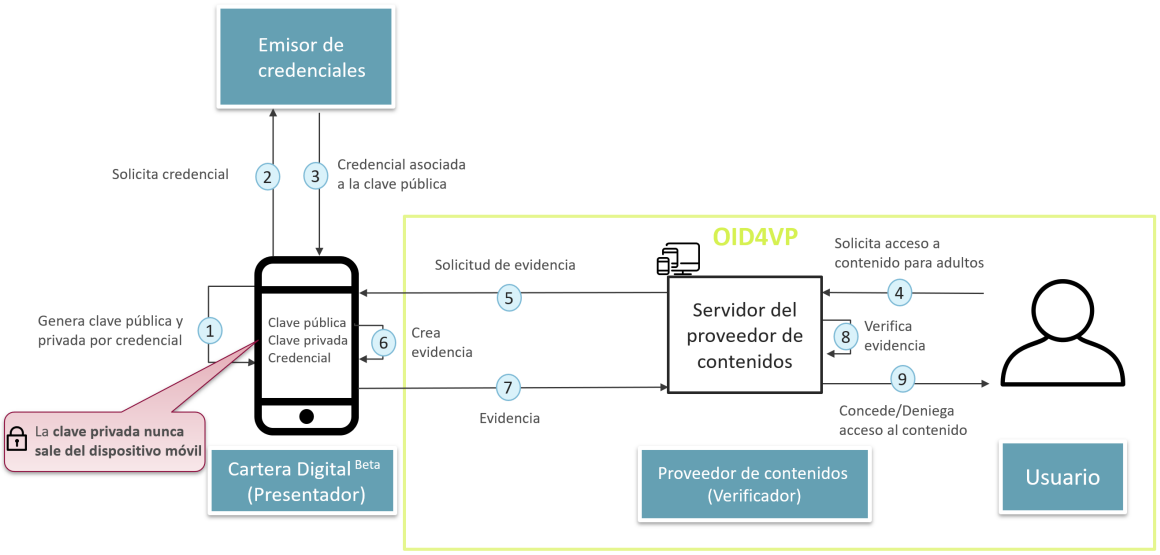


Figura 2. Flujo general: OID4VP

Para poder acometer la parte del flujo de acceso a contenido restringido para adultos, primero el usuario final debe tener una credencial verificable de mayoría de edad, es decir, previamente se debe de haber realizado el siguiente flujo de emisión:

1. Para cada credencial que se solicite el dispositivo móvil generará un par de claves, pública y privada, relacionadas entre sí matemáticamente. La clave privada nunca saldrá del dispositivo mientras que la clave pública se facilitará al emisor de credenciales en la solicitud de credencial.
2. Se solicita la credencial facilitando el DID, identificador descentralizado generado a partir de la clave pública generada previamente.
3. El emisor emite la credencial verificable asociada al DID, lo que hace al poseedor de la clave privada asociada a esa clave pública titular de la credencial. El hecho de que las credenciales se asocien a las claves públicas brinda trazabilidad puesto que se podrían correlacionar todas las credenciales que se presentan a proveedores de contenido mediante la clave pública a la que están asociadas, con el objetivo de disminuir la trazabilidad, se genera un par de claves por cada credencial de forma que la trazabilidad se reduzca únicamente a una sola credencial y resulte imposible correlacionar credenciales entre sí. En el documento "Especificación de uso credencial mayoría de edad" se explica la solución adoptada en Cartera Digital BETA para minimizar la trazabilidad.
4. Una vez se dispone de la credencial de mayoría de edad, el usuario final puede solicitar acceso al contenido restringido para adultos.

5. La solicitud de acceso del usuario final deriva en una solicitud de evidencia generada por el proveedor de contenidos que recibirá la aplicación móvil Cartera Digital[BETA] con las indicaciones sobre la credencial que se solicita, formato, algoritmos y todos los detalles necesarios para poder obtener acceso al contenido.
6. Basándose en la solicitud, Cartera Digital[BETA], genera la evidencia.
7. Se envía la evidencia al proveedor de contenidos.
8. El proveedor de contenidos verifica la evidencia.
9. Si la evidencia presentada supera las verificaciones realizadas por el proveedor de contenidos con éxito, este dará acceso al usuario final al contenido solicitado. En caso contrario, el acceso le será denegado.

2 MODELO DE DATOS

2.1 Solicitud de Evidencia

Esta sección está dedicada a la especificación del modelo de datos de la solicitud de evidencia realizada por el proveedor de contenidos a la aplicación móvil Cartera Digital [BETA] una vez que el usuario final solicita acceso al contenido restringido para adultos.

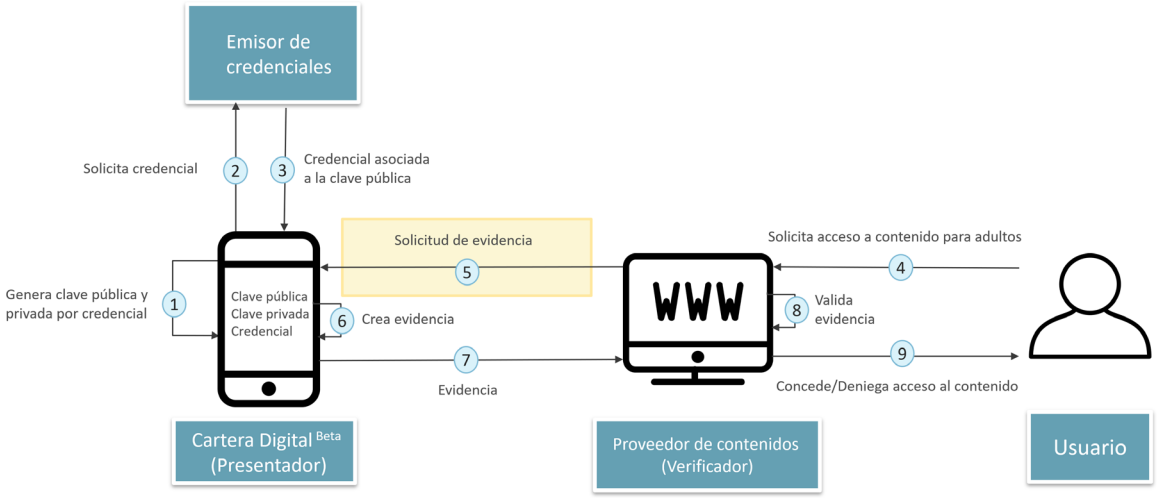


Figura 3. Flujo general: Solicitud de evidencia

La solicitud de evidencia contendrá la URI (Uniform Resource Identifier) que la aplicación móvil utilizará para obtener los parámetros de la solicitud de evidencia. La URI que referencia los datos de la solicitud se construye añadiendo los siguientes parámetros a la URL del *endpoint* de autorización utilizando la codificación *application/x-www-form-urlencoded*:

- `request_uri`
  - La URI absoluta es la URL que referencia los parámetros de la solicitud de evidencia, es decir, la URL a la que la aplicación móvil realizará una petición para obtener el objeto que contiene los parámetros de la solicitud de evidencia.
- `client_id`
  - El valor debe coincidir con el establecido en el campo `client_id` del objeto que contiene los parámetros de la solicitud de evidencia.

Esta URI referencia los parámetros comprendidos en la solicitud de evidencia, puede ser o bien un deep link, universal link o app link y no debe exceder los 521 caracteres ASCII.

El esquema de protocolo configurado en la aplicación Cartera Digital [BETA] es *ageverification* por lo que lo siguiente es un ejemplo ilustrativo de una solicitud de evidencia generada

```
ageverification://authorize? client_id=https%3A%2F%2Fwwww.todoporno.es%2F
postpresvp&request_uri=https%3A%2F%2F
wwww.todoporno.es%2Frequest.json%2FGkurKxf5T0Y-
mnPFCHqWOMiZi4VS138cQO_V7PZHAdM
```

2.2 Objeto de la solicitud de evidencia

Esta sección detalla el modelo de datos del objeto que contiene los parámetros de la solicitud de la evidencia, es decir, el modelo de datos que se devolverá cuando la aplicación móvil Cartera Digital [BETA] realice un GET a la URL facilitada en el campo `request_uri` de la solicitud de evidencia especificada en la sección previa.

```
{
    "response_type": "vp_token",
    "client_id_schema": "redirect_uri",
    "response_mode": "direct_post.jwt",
    "response_uri": "${URI de vuelta}",
    "client_id": "${response_uri}",
    "nonce": "07d54d63-7136-3ff1-11d8-f 9d17bdb0620",
    "presentation_definition": {
       "id": "32f54163-7166-48f1-93d8-f f217bdb0653",
       "format": {
              "jwt_vc": {
                     "alg": ["RS512"
              },
               "jwt_vp": {
                     "alg": ["RS512"
              },
        },
        "input_descriptors": [{
              "id": "Age over 18",
              "constrainst": {
                     "fields": [{
                            "path": [
                                   "$.credentialSubject.ageOver18"
                            ]
                     }]
              },
              "format": {
                     "jwt_vc": {
                            "alg": ["RS512"]
                     }
              }
       }]
    }
}
```

Contendrá los siguientes campos dados por el protocolo [OpenID4VP]:

- `presentation_definition`:
  - Es un objeto JSON definido en la especificación [DIF.PresentationExchange] sobre el Intercambio de Presentaciones. Son objetos que definen que pruebas solicita un verificador, en este caso, el proveedor de contenido.
  - `id`:

- Campo obligatorio. Se aconseja utilizar un identificador único, como un Universal Unique Identifier (UUID) con formato *String*.

- `format`:
  - Opcional. Objeto que indica a la aplicación móvil la configuración de formatos que el proveedor de contenidos es capaz de procesar. El proveedor de contenido puede aceptar varios formatos, dentro de los formatos establecidos en [DIF.ClaimFormatRegistry] (*jwt*, *jwt_vc*, *jwt_vp*, *ldp*, *ldp_vc* o *ldp_vp*). El siguiente es un ejemplo ilustrativo:

```
{
        "jwt": {
                "alg": ["EdDSA", "ES256K", "ES384"]
              },
        "jwt_vc": {
                "alg": ["EdDSA", "ES384"]
        },
        "jwt_vp": {
                "alg": ["EdDSA", "ES256K"]
        },
        "ldp_vc": {
                "proof_type": [
                  "JsonWebSignature2020",
                  "Ed25519Signature2018",
                  "EcdsaSecp256k1Signature2019",
                  "RsaSignature2018"
                ]
         },
        "ldp_vp": {
                "proof_type": ["Ed25519Signature2018"]
        },
        "ldp": {
                "proof_type": ["RsaSignature2018"]
        }
}
```

Es importante destacar que, todo proveedor de servicio debe soportar el algoritmo de firma RS512, dado que es este el soportado por la aplicación móvil Cartera Digital [BETA].

- `input_descriptors`:
  - Campo obligatorio. Array de objetos de tipo *Input Descriptors*, objetos que contienen los siguientes campos:
    - `id`: Obligatorio. *String* que identifica el *input descriptor*, no puede coincidir con la propiedad `id` de otro *input descriptor* contenido en la misma definición de presentación.
    - `format`: Opcional. Es idéntica a la propiedad `format` de la definición de presentación, pero puede utilizarse si se quiere restringir el formato de un input descriptor en específico, véase propiedad `format` para más detalle.
    - `constraints`: Obligatorio. Es un objeto compuesto de las siguientes propiedades:
      - `fields`: Opcional. Objeto JSON. Los campos se procesan en orden, por lo que, si se desea reducir el procesamiento comprobando las características más relevantes de la credencial, las validaciones de estos campos se deberán de

ordenar primero al implementar la solución. El campo `path` debe estar obligatoriamente presente en el objeto `fields`, es una lista de una o más expresiones *JSONPaths string*. Adicionalmente, pueden añadirse campos opcionales descritos en la especificación de intercambio de presentaciones.

        - `limit_disclosure`: Opcional. Se puede fijar como *required*, indicando que solo se pueden presentar los campos listados en `fields` o se puede fijar como *preferred* indicando que es aconsejable hacerlo.

- `client_id_scheme`:
    - Valor fijado a `redirect_uri`. Dado que el marco de confianza está basado en una lista blanca en la que se identificará a los proveedores de contenido en base a la URI a la que se envía la evidencia, será esta URI la que se utilizará como `client_id` en la solicitud de evidencia.
- `nonce`:
    - Obligatorio. *String* que se utiliza para prevenir ataques de réplica, un atacante podría intentar insertar la presentación verificable incluida en una evidencia en otra evidencia, mediante el `nonce` estas se podrán presentar una única vez. Servirá también para vincular la solicitud de evidencia con la evidencia presentada por la aplicación móvil Cartera Digital [BETA].
- `state`:
    - Opcional. Gestiona la sesión, le permite al proveedor de contenidos vincular la solicitud con la evidencia. La sesión estará vinculada con la cookie del navegador desde el que se ha solicitado el acceso, de forma que, si se verifica la evidencia se autorizará esa cookie para visualizar el contenido. Por tanto, si un tercero presenta la evidencia, la cookie autorizada será la del navegador del usuario que la solicitó y no la del usuario que presenta la evidencia.
- `response_mode`:
    - Valor fijado a `direct_post`. Esto permite a la aplicación móvil enviar la evidencia mediante una petición HTTPS POST, soluciona problemas como el exceder el límite de tamaño de la URL o el no poder enviar la respuesta mediante redirección al verificador dado que el proveedor de contenido y la aplicación móvil están en dispositivos diferentes. Los parámetros de la respuesta de autorización serán codificados en el cuerpo utilizando el formato *application/x-www-form-urlencoded*.
- `response_uri`: Obligatorio en el caso de que el campo `response_mode` se establezca como `direct_post`. La URI a la que la aplicación móvil debe enviar la evidencia, la URI que se utiliza como identificador del proveedor de contenidos en la lista blanca de proveedores de contenido.

Contendrá los siguientes campos heredados de la norma OAuth2.0:

- `response_type`

- Campo obligatorio fijado a `vp_token`, según establece el protocolo [OpenID4VP].

- `client_id`
  - Campo obligatorio cuyo objetivo es identificar al cliente OAuth2.0, es decir, al proveedor de contenido. Dado que el `client_id_schema` se establece a `redirect_uri` y teniendo en cuenta que no se utilizará una URI de redirección por evitar posibles problemas, sino que se utilizará el modo de respuesta `direct_post` que permite enviar la respuesta mediante una petición HTTP POST, *el campo* `client_id` será igual a el campo `response_uri`.

2.3 Evidencia

Una vez que la aplicación móvil ha recibido la solicitud de evidencia, procede a generar la evidencia siguiendo los requisitos especificados en la solicitud.

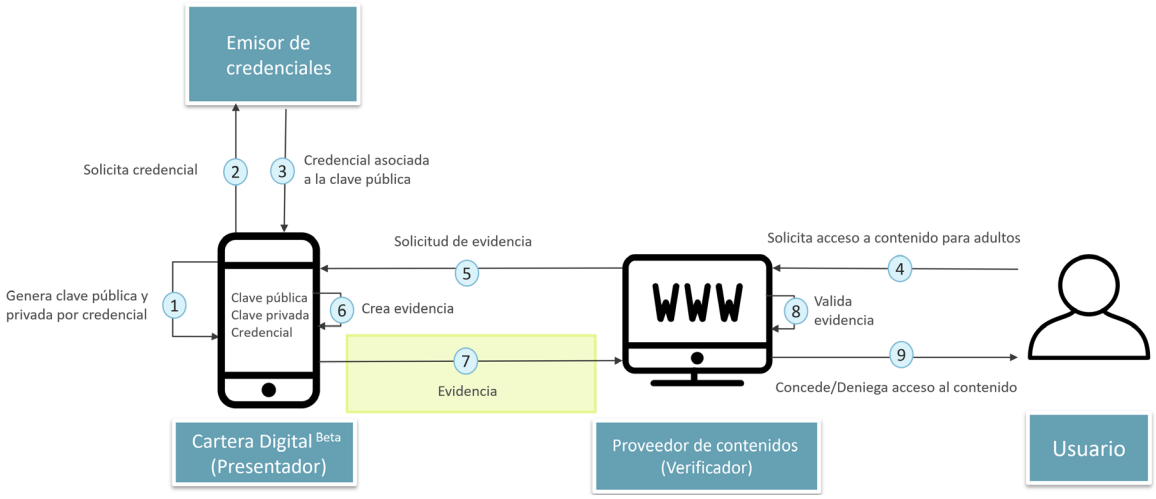


Figura 4. Flujo general: Evidencia

La evidencia es un JWT (JSON web Token) compuesto por, en este caso de uso, una única presentación verificable que contiene una credencial verificable de mayoría de edad, tal y como se muestra en el siguiente ejemplo ilustrativo.

Figura 5. Evidencia

En respuesta a la solicitud de la evidencia que le ha sido proporcionada a la aplicación móvil por el proveedor de contenidos, se le devolverá al proveedor de contenidos la evidencia cuyo cuerpo sigue el modelo de datos detallado a continuación.

```
{
        "vp_token": {
           "@context": "https://www.w3.org/ns/credentials/v2",
           "id":"data:application/vp+ld+json+jwt;${presentacionVerificableJWT}",
           "type": "EnvelopedVerifiablePresentation"
        },
        "presentation_submission": {
                "id": "a30e3b91-fb77-4d22-95fa-871689c322e2",
                "definition_id": "32f54163-7166-48f1-93d8-f f217bdb0653",
                "descriptor_map": [ {
                        "id": "Age over 18",
                        "format": "jwt_vc",
                        "path": "$.verifiableCredential[0]"
                }]
        },
        "nonce": "07d54d63-7136-3ff1-11d8-f 9d17bdb0620"
}
```

Dado que el campo `response_type` establecido en la solicitud de autorización es `vp_token`, la respuesta de autorización contendrá los siguientes parámetros:

- `vp_token`
  - Obligatorio. Dado que se va a incluir una única Presentación Verificable, será un objeto JSON que incluye la misma. En caso de que se tratase de múltiples Presentaciones verificables, sería una lista de objetos JSON, cada uno de ellos incluirá una de las Presentaciones Verificables.
- `presentation_submission`
  - Obligatorio. Objeto que contiene los siguientes campos:

        - `id`: Obligatorio, debe ser un identificador único, como, por ejemplo, un UUID.
        - `definition_id`: Obligatorio. El campo `id` de la definición de presentación.
        - `descriptor_map`: Lista de objetos *Input Descriptor Mapping* compuestos de los siguientes campos:
            - `id`: Obligatorio. String que coincide con el campo id de la propiedad *Input Descriptor* de la definición de presentación en la que se basa la respuesta.
            - `format`: Obligatorio. Denota el formato de las afirmaciones y debe ser una de las propuestas por DIF.
            - `path`: Obligatorio. Debe de ser una expresión en formato *string* de tipo *JSONpath*. Indica el atributo respecto al *Input Descriptor* identificado.
- `nonce`:
    - *String*. Debe coincidir con el `nonce` facilitado en los parámetros de la solicitud de evidencia.
- `state`:
    - Opcional. *String*. Debe coincidir con el campo `state` facilitado en los parámetros de la solicitud de autorización. En este caso de uso no será necesario puesto que se utilizará el campo `nonce` para vincular la solicitud con la evidencia.
- `exp`: Obligatorio. *NumericDate*. Fecha de expiración, tras ese momento no debe de aceptarse la evidencia.
- `aud`: Obligatorio. *String*. Identifica al receptor para el que se ha generado el JWT. Se utilizará el valor que venga en el campo `response_uri` de la solicitud puesto que este es el identificador del proveedor de contenido.

A pesar de que los atributos `exp` y `aud` estén definidos en la RFC7519 como opcionales, serán obligatorios en la presente especificación dada la relevancia de estos para la solución del caso de uso de mayoría de edad.

La evidencia que recibe el proveedor de contenidos es un JWT cuyo cuerpo se corresponde con el descrito y asegurado mediante una firma realizad con la clave privada que reside en el dispositivo móvil del usuario final. Se trata de una prueba que demuestran el control sobre la clave privada asociada a la clave pública del titular de la credencial. El algoritmo de firma será ECDSA con SHA256 (ecdsaSignatureMessageX962SHA256). La curva P256. La cabecera del JWT contendrá las siguientes propiedades:

- `alg`: *String*. Obligatorio. Valor fijado a ES256.

2.4 Presentación Verificable

La evidencia detallada previamente contiene una Presentación Verificable, en el campo `vp_token`, generada por la aplicación móvil Cartera Digital [BETA] siguiendo el modelo de datos de W3C:

```
{
   "@context": "https://www.w3.org/ns/credentials/v2",
   "id":"data:application/vp+ld+json+jwt;${presentacionVerificableJWT}”,
   "type": "EnvelopedVerifiablePresentation"
}
```

Se trata de un JSON compuesto por las siguientes propiedades:

- `@context`
  - Valor fijado a *https://www.w3.org/ns/credentials/v2*.
- `id`
  - Representa una presentación verificable asegurada mediante el mecanismo *JOSE (JavaScript Object Signing and Encryption)*. El formato del campo está definido en el esquema URL data y se cumplimentará de la siguiente manera:

```
data:application/vp+ld+json+jwt;${presentaciónVerificableJWT}
```

- `type`
  - Valor fijado a *EnvelopedVerifiablePresentation* .

Se debe extraer y decodificar el JWT incluido en el campo `id` de la Presentación Verificable, JSON que contiene las propiedades que se detallan a continuación y está asegurado mediante cifrado con la clave privada del titular de la Credencial Verificable que se incluye, de forma que, solo el titular de las credenciales pueda presentar estas.

```
{
        "id": "urn:uuid:00000000-0000-0000-0000-000000000000",

"type":  [
                "VerifiablePresentation"
        ],
        "verifiableCredential":[{
                "@context" : "https://www.w3.org/ns/credentials/v2",
                "id":"data:application/vc+ld+json+jwt;${credencialVerificableJWT}”,
                "type": "EnvelopedVerifiableCredential"
        }],
        "holder":"did:key:z2dmzD81cgPx8Vki7JbuuMmFYrWPgYoytykUZ3eyqht1j9KbrSNto1XXZFRD5StnZ
        PJ1tLKTc39AJ3Ae1EW99bJhMpXJgEq8BaqpX2UCrbsxG9fDpXKLFswiEdJisHwMqhTWrMUTe7pHH8Vo3Zkt
        nujZVd7HuTCwjrvEv4m1r8yTKQt35e"
}
```

Estando esta conformada por los siguientes campos:

- `id`
  - Campo reservado para su uso futuro. Valor fijado a *urn:uuid:00000000-0000-0000-0000-000000000000*.
- `type`

- Lista de *Strings*. Valor fijado a *VerifiablePresentation.*

- `verifiableCredential`
  - `@context`
    - Valor fijado a *https://www.w3.org/ns/credentials/v2* .
  - `type`
    - Valor fijado a *EnvelopedVerifiableCredential.*
  - `id`
    - Representa una credencial verificable asegurada mediante el mecanismo JOSE. El formato del campo está definido en la norma [RFC2397] y se cumplimentará de la siguiente manera:

```
data:application/vc+ld+json+sd-jwt;${credencialVerificableJWT}
```

- `holder`
  - DID del titular de la Credencial Verificable.

2.5 Credencial Verificable

La Presentación Verificable detallada en la sección previa contiene, en el campo `verifiableCredential`, la Credencial Verificable de mayoría de edad que el proveedor de contenidos deberá verificar generada por el emisor de la credencial siguiendo el modelo de datos de W3C.

Se debe extraer y decodificar el JWT incluido en el campo `id` de la Credencial Verificable, JSON que contiene las propiedades que se detallan a continuación y está asegurado mediante la firma del emisor de la Credencial Verificable, de forma que, el proveedor de contenido restringido para adultos pueda verificar a la entidad emisora de esta, tal y como se muestra en el siguiente ejemplo ilustrativo.

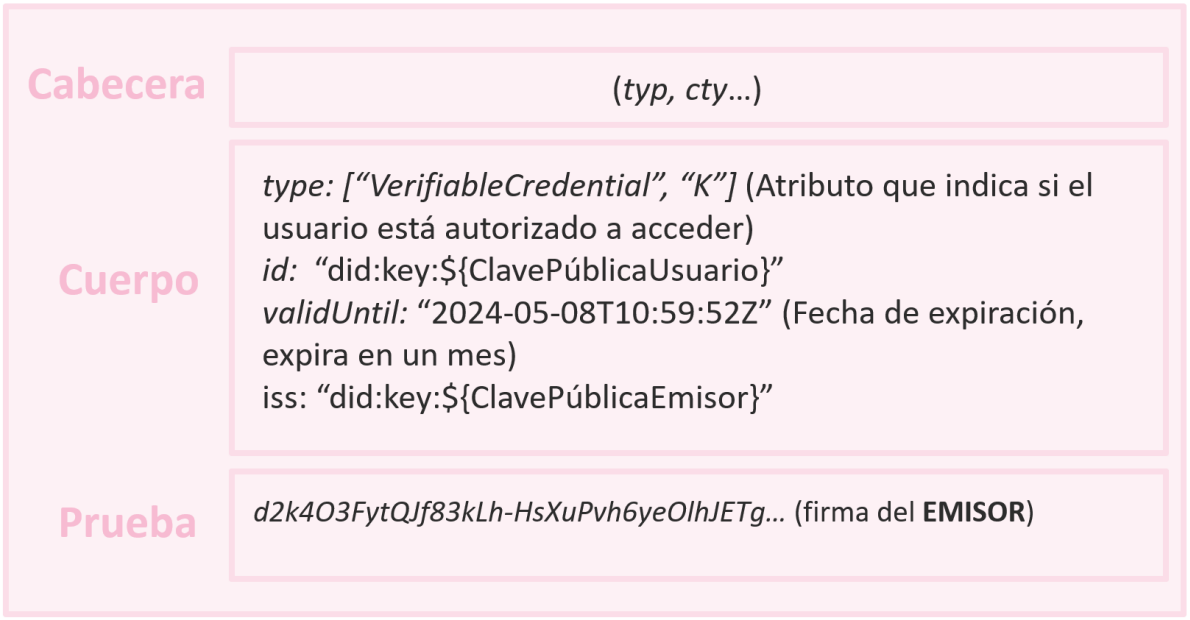


Figura 6. Credencial verificable de tipo K

La cabecera incluirá al menos los siguientes campos:


```
{
        "x5c":[ "MIIHrjCCBpagAwIBAgIQft7+0deAk5JhFQlRBPK9njANBgkqhkiG9w0BAQsFADBHMQswCQYDVQ
        QGEwJFUzERMA8GA1UECgwIRk5NVC1SQ00xJTAjBgNVBAsMHEFDIENvbXBvbmVudGVzIEluZm9ybcOhdGljb
        3MwHhcNMjEwODEyMTE0MzEzWhcNMjQwODEyMTE0MzEyWjCB7DELMAkGA1UEBhMCRVMxDzANBgNVBAcMBk1B
        RFJJRDFCMEAGA1UECgw5TUlOSVNURVJJTyBERSBBU1VOVE9TIEVDT05PTUlDT1MgWSBUUkFOU0ZPUk1BQ0l
        PTiBESUdJVEFMMTUwMwYDVQQLDCxTRUNSRVRBUklBIEdFTkVSQUwgREUgQURNSU5JU1RSQUNJT04gRElHSV
        RBTDESMBAGA1UEBRMJUzI4MDA1NjhEMRgwFgYDVQRhDA9WQVRFUy1TMjgwMDU2OEQxIzAhBgNVBAMMGlNFT
        ExPIEVOVElEQUQgU0dBRCBQUlVFQkFTMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAvdOB/mRK
        zFJSZKbDrBvZoomo+Yuc+IKr9uiYIRvTgz+yEqucVRejRhMjsIh1MiT1GlRb+V9iST6pRj7t/aS8H6Sqziz
        Iy756TgsJz8GVRbOfX2A1XCN5QK0Fo96HmcADVj01M18ze+VQz7YG0Q/onbdx5IZwNncxOn3e0fGw2TEb85
        wuynhBND3ci2341+zh/zhcHEd0rMXv6NJkTi2DiS5aVx8/ou4LjFum9HmFBrIOfbVv8r+Q5W1q494HlRqGq
        /rnTyGisq3YAC2iEE/ctJJf86dza/b08lb9yFT+WBmW6Zs5Aa3CvI5+dnGrEJk/O3v+JiBMaDaMzkwFZ8Nk
        CQIDAQABo4ID7jCCA+owDAYDVR0TAQH/BAIwADCBgQYIKwYBBQUHAQEEdTBzMDsGCCsGAQUFBzABhi9odHR
        wOi8vb2NzcGNvbXAuY2VydC5mbm10LmVzL29jc3AvT2NzcFJ1c3BvbmRlcjA0BggrBgEFBQcwAoYoaHR0cD
        ovL3d3dy5jZXJ0LmZubXQuZXMvY2VydHMvQUNDT01QLmNydDCCATQGA1UdIASCASswggEnMIIBGAYKKwYBB
        AGsZgMJEzCCAQgwKQYIKwYBBQUHAgEWHWh0dHA6Ly93d3cuY2VydC5mbm10LmVzL2RwY3MvMIHaBggrBgEF
        BQcCAjCBzQyBykNlcnRpZmljYWRvIGN1YWxpZmljYWRvIGRlIHNlbGxvIGVsZWN0csOzbmljbyBzZWfDum4
        gcmVnbGFtZW50byBldXJvcGVvIGVJREFTLiBTdWpldG8gYSBsYXMgY29uZGljaW9uZXMgZGUgdXNvIGV4cH
        Vlc3RhcyBlbiBsYSBEUEMgZGUgRk5NVC1SQ00gY29uIE5JRjogUTI4MjYwMDQtSiAoQy9Kb3JnZSBKdWFuI
        DEwNi0yODAwOS1NYWRyaWQtRXNwYcOxYSkwCQYHBACL7EABATA4BgNVHREEMTAvpC0wKzEpMCcGCSsGAQQB
        rGYBCAwaU0VMTE8gRU5USURBRCBTR0FEIFBSVUVCQVMwDgYDVR0PAQH/BAQDAgXgMB0GA1UdDgQWBBR/WVm
        pRLawlHxewC4lJsqtEgQENzCBsAYIKwYBBQUHAQMEgaMwgaAwCAYGBACORgEBMAsGBgQAjkYBAwIBDzATBg
        YEAI5GAQYwCQYHBACORgEGAjByBgYEAI5GAQUwaDAyFixodHRwczovL3d3dy5jZXJ0LmZubXQuZXMvcGRzL
        1BEU19DT01QX2VzLnBkZhMCZXMwMhYsaHR0cHM6Ly93d3cuY2VydC5mbm10LmVzL3Bkcy9QRFNfQ09NUF9l
        bi5wZGYTAmVuMB8GA1UdIwQYMBaAFBn4WC8U1qbMmwSYCA1M16sAp4NlMIHgBgNVHR8EgdgwgdUwgdKggc+
        ggcyGgZ5sZGFwOi8vbGRhcGNvbXAuY2VydC5mbm10LmVzL0NOPUNSTDEsT1U9QUMlMjBDb21wb25lbnRlcy
        UyMEluZm9ybWF0aWNvcyxPPUZOTVQtUkNNLEM9RVM/Y2VydGlmaWNhdGVSZXZvY2F0aW9uTGlzdDtiaW5hc
        nk/YmFzZT9vYmplY3RjbGFzcz1jUkxEaXN0cmlidXRpb25Qb2ludIYpaHR0cDovL3d3dy5jZXJ0LmZubXQu
        ZXMvY3Jsc2NvbXAvQ1JMMS5jcmwwDQYJKoZIhvcNAQELBQADggEBACSyNnj3Kd05oB7fScFj+2sy5CKtHry
        iCIwrlLAGHWS0rik20WSrx1+hgkRJxgsKvqH5mxDzddWfg93+NwEP6JCHPMtPqfWMaLHGuHQfRYxKJx00ak
        ufhgH2knpAEFJhB5SGEU0gQuRSijvF5+a+VhX9evLt3RwnGYIwcPo34MTlr+PHR7PTgzjGxYn/8g8etrOeG
        e3eZAIp0YNTDSUA1BLSdhDjM6zP7DdTOFJGcfu3pJAx3EpuEBaSQlyxc2FWDqWrP6xPKgz5VdqsJ4Zlekka
        0SRI+ub/dNQ4I44Pj+qs15FWSUF3Mt+VBNizg2omHpdcccB2xpgBsBN5k0GzT50="],
        "kid":
        "MIIBCgKCAQEAvdOB/mRKzFJSZKbDrBvZoomo+Yuc+IKr9uiYIRvTgz+yEqucVRejRhMjsIh1MiT1GlRb+V
        9iST6pRj7t/aS8H6SqzizIy756TgsJz8GVRbOfX2A1XCN5QK0Fo96HmcADVj01M18ze+VQz7YG0Q/onbdx5
        IZwNncxOn3e0fGw2TEb85wuynhBND3ci2341+zh/zhcHEd0rMXv6NJkTi2DiS5aVx8/ou4LjFum9HmFBrIO
        fbVv8r+Q5W1q494HlRqGq/rnTyGisq3YAC2iEE/ctJJf86dza/b08lb9yFT+WBmW6Zs5Aa3CvI5+dnGrEJk
        /O3v+JiBMaDaMzkwFZ8NkCQIDAQAB",
        "alg": "RS512"
}
```


- `x5c`
  - Contiene el certificado de clave pública X.509 o la cadena de certificados correspondientes a la clave utilizada para firmar el cuerpo de la credencial. Campo definido en el punto 4.1.6 de la norma [RFC7515].
- `kid`
  - Campo definido en el punto 4.1.4 de la norma [RFC7515]. Los proveedores de contenido ignorarán este campo, la clave pública se extraerá del certificado presente en el campo `x5c`.
- `alg`
  - El algoritmo de firma utilizado por el emisor de la credencial. Valor fijado a *RS512*. Campo descrito en la sección 3.1 de la norma [RFC7518].

El cuerpo de la credencial está compuesto de los siguientes campos:

```
{
    "@context": ["https://www.w3.org/ns/credentials/v2"],
    "id": "urn:uuid:00000000-0000-0000-0000-000000000000",
    "type": ["VerifiableCredential", "K"],
    "credentialSubject": {
        "id": "did:key:${ClavePúblicaUsuario}"
    },
    "validFrom": "2023-01-01T00:00:00Z",
    "validUntil": "2024-05-08T10:59:52Z",
    "issuer": "did:key:${ClavePúblicaEmisor}",
}
```

- `@context`
  - Valor fijado a *https://www.w3.org/ns/credentials/v2* .
- `type`
  - Valor fijado a *["VerifiableCredential", "K"].* El tipo K indica que se trata de una credencial que acredita al usuario como mayor de edad, y, por tanto, se debe conceder el acceso al contenido restringido para adultos.
- `credentialSubject`
  - JSON que expresa atributos del usuario formado por la siguiente propiedad:
    - `id`: DID del titular de la credencial.
- `validFrom`
  - Es un XMLSCHEMA11-2 *dateTimeStamp string* que representa la fecha y el momento en el que la credencial comienza a ser válida.
- `validUntil`
  - Es un XMLSCHEMA11-2 *dateTimeStamp string* que representa la fecha y el momento en el que la credencial deja de ser válida.
- `issuer`
  - DID de tipo key del emisor de la credencial especificado en la norma [DID-Key].

Por último, el cuerpo de la credencial estará asegurado siguiendo el formato de firmas JOSE especificado en la norma [RFC7515] con el algoritmo de firma *RS512*.

2.6 Listas blancas

El marco de confianza de la presente solución está basado en listas blancas en las que se registran todas las entidades consideradas de confianza para realizar una acción, como, por ejemplo, emitir cierto tipo de credenciales. La lista blanca de aplicaciones móviles para la presentación de evidencias de mayoría de edad será un espacio público, en el que, de cara a todos los actores, se publicitarán las aplicaciones disponibles, la primera de ellas será Cartera Digital [BETA]. Además, existirán dos listas blancas, una de emisores de confianza y otra de proveedores de contenido en la que se acreditará a estos para emitir y solicitar credenciales de un determinado tipo respectivamente. Estas dos listas blancas son consultadas y verificadas automáticamente por los actores involucrados en el proceso y residirán en servidores públicos.

Ambas listas blancas son un fichero JSON firmado electrónicamente siguiendo el formato de firmas JOSE especificado en la norma [RFC7515], usando para ello el Certificado de Sello del gestor de la lista blanca. La cabecera contendrá los siguientes campos:

- `x5c`: definido en el punto 4.1.6 de la citada norma [RFC7515].
- `kid`: Identificador de la clave pública definido en la sección 4.1.4 [RFC7515].
- `alg`: El algoritmo de firma utilizado. Valor fijado a RS512, tal y como se describe en la norma [RFC7518], punto 3.1.

2.6.1 Lista blanca de emisores

A continuación, se muestra un modelo con todas las etiquetas habilitadas de la lista blanca contra la que se debe validar a los emisores de las credenciales verificables:

```
{
  "trustIssuersStatusList": {
    "id": "TISL20240326",
    "nextUpdate": {
      "dateTime": "2024-09-22T00:00:00Z"
    },
    "distributionPoints": {
      "uri": [
        "URI de publicación de la lista blanca"
      ]
    },
    "schemeInformation": {
      "tislVersionIdentifier": "5",
      "schemeName": [
        {
          "lang": "en",
          "text": "EN:Trusted list including information related to trusted issuers."
        },
        {
          "lang": "es",
          "text": "ES:Lista de confianza que incluye información relacionada con los emisores de confianza."
        }
      ]
    },
    "trustIssuerList": [
      {
        "issuerName": [
          {
            "lang": "es",
            "text": "Organismo responsable de la emisión"
          },
          {
            "lang": "en",
            "text": "Issuing authority"
          }
        ],
        "issuerAddress": {
          "postalAddress": [
            {
              "streetAddress": "Registered address of the issuer",
              "locality": "Madrid",
              "stateOrProvince": "Madrid",
              "postalCode": "28020",
              "countryName": "ES",
              "lang": "en"
            },
            {
              "streetAddress": "Domicilio social del emisor",
              "locality": "Madrid",
              "stateOrProvince": "Madrid",
              "postalCode": "28020",
              "countryName": "ES",
              "lang": "es"
            }
          ],
          "electronicAddress": [
            {
              "lang": "es",
              "text": "Portal web del emisor"
            },
            {
              "lang": "es",
              "text": "Dirección de correo del emisor"
            }
          ]
        },
        "schemeTerritory": "ES",
        "authorizedToIssue": [
          "K",
          "UD"
        ],
```

```
      "policyOrLegalNotice": {
        "tislLegalNotice": [
          {
            "lang": "en",
            "text": "The applicable legal framework for the present trusted list"
          },
          {
            "lang": "es",
            "text": "El marco jurídico aplicable a la presente lista de confianza"
          }
        ]
      },
      "serviceDigitalIdentities": [
        {
          "digitalId": {
            "did": "did:key:Z2xvYmFsLmNvbS9xdmJlY2FnMi5jcnQwLQYIKwYBBQUHMAGGIWh0dHA6Ly91dy",
            "x509Certificate":
              "MIIHLjCCBRagAwIBAgIUP4a8WPhZN++7VkfYyuAcDGmTglEwDQYJKoZIhvcNAQELBQAwczELMAkGA1UEBhMCQkUxGTAXBgNVBGEMEE5UUkJFLTA1Mzc2OTgzMTgx
              IDAeBgNVBAoMF1F1b1ZhZGlzIFRydXN0bGluayBCVkJBMScwJQYDVQQDDB5RdW9WYWRpcyBCZWxnaXVtIElzc3VpbmcgQ0EgRzIwHhcNMjEwMTIxMDg0ODM4WhcNM
              jQwMTIxMDg1ODAwWjCBwDELMAkGA1UEBhMCQkUxLDAqBgNVBAsMI0RpcmVjdG9yYXRlUdlbmVyYWwgZm9yIEluZm9ybWF0aWNzMRkwFwYDVQRhDBBOVFJCRS0wOT
              Q5MzgzMzQyMUowSAYDVQQKDEFDT01NSVNTSU9OIERFIEwnVU5JT04gRVVST1BFRU5ORSAtIENPTU1JU1NJRSBWQU4gREUgRVVST1BFU0UgVU5JRTEcMBoGA1UEAww
              TRXVyb3BlYW4gQ29tbWlzc2lvbjCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAMCq40SQNaqRkbc5nsdPjLKyxghYzBLJCmeN3/ndD5mMzhy9tsP8WOps
              fi8CnMD5tasEEmS5Fj73DQ+sL9PIyfmLzBJVhuCyzwKpDiFAhAa+ec/LTyVRa6+oCh918ZUcPDVXpCtGA3t6WS0pCgrIJNkZequ2yIwUZxiyVOACYmkxmWhWnGe7L
              RaLv80PCI0QERL2BzV+mKb2tQylUphPPI8hd9LG9EwCvSSXv0tb4rK1C+lIcxuYsui7dNyfw9UHDBnuI5452c3O18N5+GSI9uiEmOI8s++fbubn61RskePhk3ISUR
              PGIAeDKV9pCZezBGysq5IlNkC1C4AtYS2eKFcCAwEAAaOCAmowggJmMB8GA1UdIwQYMBaAFIfJvDGXEnpzu37APUVRtAEllVGrMHcGCCsGAQUFBwEBBGswaTA4Bgg
              rBgEFBQcwAoYsaHR0cDovL3RydXN0LnF1b3ZhZGlzZ2xvYmFsLmNvbS9xdmJlY2FnMi5jcnQwLQYIKwYBBQUHMAGGIWh0dHA6Ly91dy5vY3NwLnF1b3ZhZGlzZ2xv
              YmFsLmNvbTBaBgNVHSAEUzBRMEQGCisGAQQBvlgBgxAwNjA0BggrBgEFBQcCARYoaHR0cDovL3d3dy5xdW92YWRpc2dsb2JhbC5jb20vcmVwb3NpdG9yeTAJBgcEA
              IvsQAEDMCkGA1UdJQQiMCAGCCsGAQUFBwMCBggrBgEFBQcDBAYKKwYBBAGCNwoDDDCBiwYIKwYBBQUHAQMEfzB9MBUGCCsGAQUFBwsCMAkGBwQAi+xJAQIwCAYGBA
              CORgEBMAgGBgQAjkYBBDATBgYEAI5GAQYwCQYHBACORgEGAjA7BgYEAI5GAQUwMTAvFilodHRwczovL3d3dy5xdW92YWRpc2dsb2JhbC5jb20vcmVwb3NpdG9yeRM
              CZW4wOwYDVR0fBDQwMjAwoC6gLIYqaHR0cDovL2NybC5xdW92YWRpc2dsb2JhbC5jb20vcXZiZWNhZzIuY3JsMB0GA1UdDgQWBBQspnrB2Vq7SM99egtZxs2jZv1Q
              OjAOBgNVHQ8BAf8EBAMCBsAwEwYKKoZIhvcvAQEJAgQFMAMCAQEwNAYKKoZIhvcvAQEJAQQmMCQCAQGGH2h0dHA6Ly90cy5xdW92YWRpc2dsb2JhbC5jb20vYmUwD
              QYJKoZIhvcNAQELBQADggIBACGQDYsjrM3nsKSpiW1SqzBlZOkvuiphSLNhv5o5V1W+maYLtXYH/fMadrP0+K2X5XB5e+8yMw4Sbrvxv4LRR/CQ2WX03KjYGcqdcI
              UAH+A/8phwA7eLyKd3ip8k8ukT9kUKMJxbqVR5O8x2vOlAM9VFirOenOI+MufFDSaNKIJKt+PDhcSxDyzjs2MRKHAJcTCs23T8fB6+Qvw3uKUgniVg4MTLFvWpL8b
              vMXX7QUyvsScskUCns918hHA4Ett7fxdWXJFFJuuculChJIf8jbXZQ/8SAoecPyADnJw0C4uGNI/zc9/DWqMSMMytFwGeOpQb0/pukdmNe3sai8SUQj4+vyhkCHKM
              D27cak9buDO7cu5yhKEuLCERTga2QvbGSeiFAa6KuyPGY1Ad9AOuz5KPyyfVG0h31FvNL0MnAib1IpZr7mPQgWOsEX9ipgQXoE2Rh11yydTiKwmyTTrp/uMOJp+6U
              D06IdgPkHmqnqaKkqzGu6RKiHLqYWX5qhWXt0EatHQ0YY+92B03cDTLMgjPHKY+wN6qPfetz0oQNmort+G2HIXb9P+Q43sqMjXpQY/I3YmM0VBmcYWUon11BOavnG
              GOCVVPBQFKU6Kt7r69li1CmfA7KgK7Jn2o/jb8BupWUPGej++MkkrFDS0T6tmNH9ZDteyO3dSyOcl1IHcs"
          }
        },
        {
          "digitalId": {
            "did": "did:key:SmgAwIBAgIUCowT7Q98oPLpaLF0FnUX3eobcgYwDQYJKoZIhvcNAQELBQAwczE",
            "x509Certificate":
              "MIIHQTCCBSmgAwIBAgIUCowT7Q98oPLpaLF0FnUX3eobcgYwDQYJKoZIhvcNAQELBQAwczELMAkGA1UEBhMCQkUxGTAXBgNVBGEMEE5UUkJFLTA1Mzc2OTgzMTgx
              IDAeBgNVBAoMF1F1b1ZhZGlzIFRydXN0bGluayBCVkJBMScwJQYDVQQDDB5RdW9WYWRpcyBCZWxnaXVtIElzc3VpbmcgQ0EgRzIwHhcNMjEwMTIxMDg0ODIzWhcNM
              jQwMTIxMDg1ODAwWjCB0zELMAkGA1UEBhMCQkUxPzA9BgNVBAsMNkRHIGZvciBDb21tdW5pY2F0aW9ucyBOZXR3b3JrcywgQ29udGVudCBhbmQgVGVjaG5vbG9neT
              EZMBcGA1UEYQwQTlRSQkUtMDk0OTM4MzM0MjFKMEgGA1UECgxBQ09NTUlTU0lPTiBERSBMJ1VOSU9OIEVVUk9QRUVOTkUgLSBDT01NSVNTSUUgVkFOIERFIEVVUk9
              QRVNFIFVOSUUxHDAaBgNVBAMME0V1cm9wZWFuIENvbW1pc3Npb24wggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQDtD1bnPz0vyI0gxH+Ye/UEyssx1TWy
              XmT2SjX8T0oR36yu+FvEI1YB7q7CGA2JNUxKdOPgFi4MZxj9vOoMwqR1+cbPDNzXsdzINLZCDy3mrPt3hfqmijhmqmwwXSQnO2+uDce+BYGXL74oWnqJTc+GT6SjY
              QMyeXvyff1JU4pC+QL53tcpcNf7dO5EJzHAfLDp8HILebtafvWAQA9+Psf3BT/0lHQZ1nqnyMfjBntCBG6/Cj2mtM44vldCKtRDdGno7MT5lxTvwoIcEA5Lupbdu9
              sudHsJT4iOub8QshI3bjHqP7n7q+OsN5RX5X/bmIG033j7Yn+DfwCKwE4eKsZHAgMBAAGjggJqMIICZjAfBgNVHSMEGDAWgBSHybwxlxJ6c7t+wD1FUbQBJZVRqzB
              3BggrBgEFBQcBAQRrMGkwOAYIKwYBBQUHMAKGLGh0dHA6Ly90cnVzdC5xdW92YWRpc2dsb2JhbC5jb20vcXZiZWNhZzIuY3J0MC0GCCsGAQUFBzABhiFodHRwOi8v
              dXcub2NzcC5xdW92YWRpc2dsb2JhbC5jb20wWgYDVR0gBFMwUTBEBgorBgEEAb5YAYMQMDYwNAYIKwYBBQUHAgEWKGh0dHA6Ly93d3cucXVvdmFkaXNnbG9iYWwuY
              29tL3JlcG9zaXRvcnkwCQYHBACL7EABAzApBgNVHSUEIjAgBggrBgEFBQcDAgYIKwYBBQUHAwQGCisGAQQBgjcKAwwwgYsGCCsGAQUFBwEDBH8wfTAVBggrBgEFBQ
              cLAjAJBgcEAIvsSQECMAgGBgQAjkYBATAIBgYEAI5GAQQwEwYGBACORgEGMAkGBwQAjkYBBgIwOwYGBACORgEFMDEwLxYpaHR0cHM6Ly93d3cucXVvdmFkaXNnbG9
              iYWwuY29tL3JlcG9zaXRvcnkTAmVuMDsGA1UdHwQ0MDIwMKAuoCyGKmh0dHA6Ly9jcmwucXVvdmFkaXNnbG9iYWwuY29tL3F2YmVjYWcyLmNybDAdBgNVHQ4EFgQU
              SiKJPwxZvli51U75EfhdERrDf98wDgYDVR0PAQH/BAQDAgbAMBMGCiqGSIb3LwEBCQIEBTADAgEBMDQGCiqGSIb3LwEBCQEEJjAkAgEBhh9odHRwOi8vdHMucXVvd
              mFkaXNnbG9iYWwuY29tL2JlMA0GCSqGSIb3DQEBCwUAA4ICAQCWLVCxzkKy/s+8kRvxCPvTlExlWkNGsrhCO+PXsh1UgWMdJCpc33OzL1GegNMSL9f3Hw6gqjXV+m
              JqtTmvrP1ZvXCagv2LRLeyxSYAjlvjwYkaraddydS8Or86OyfA0OIs1UruPe0lClb6rBKwqqx7LvXotk7XvPYQo0HrP57pCqlp4MIiNEpGeW3gRwFq8IBQcecO3k0
              09ad+tuTSB5I+o4t/7kEd//6voViy7d/EmlOgUZzXsJ24NV4pFAmyc2FlkEvYOqw8YvEsOV2hluvcG3wmkMEF/SY4X+kvpw+Ph7JQdWpWGlozzAFVppkIIrSM2Sxw
              2QXFQZWwah210HiVY0Ax4VkQbRJQNYmFWVizixPONWtc3PIfMz3/TvD9ZbExFfoQzVYXZPmtK2aQjG3CH2bJcefvS18YdR169au8yNKsPvlLaRa1k7Qv6Hv8iE3Gt
              QmszDwnQHlGrbxdft306gOFl8/U0ctyIf2YkG6J/uTnYiiibeDoHK6wBo3BfCJQ2X7/Iab3971k1wSCwe39w4x3Xw72HL9E3czRkcfpg+uTxt5jSJ7HL7BRdMD34y
              sjQ4gOV8h6NU4z+QQ3W4sZTQkFqHGHKZKt2dTrCjhE+I1hULvo4274heBr/5F0DBAXRi9lHBCBUV5iJDm90bxjHjKuvj893WLEFW3WPkWlRw=="
          }
        }
      ]
    }
  ]
}
```


Figura 7. Lista blanca de emisores de credenciales verificables

- `trustIssuersStatusList`: Elemento raíz que contiene toda la lista de emisores de confianza.
- `id`: Identificador único de la lista de emisores de confianza.
- `nextUpdate`: Fecha y hora en la que se espera que la lista de emisores de confianza se actualice.
- `distributionPoints`: Puntos de distribución donde se puede obtener la lista de emisores de confianza.
    - URI: Dirección URL donde se puede obtener la lista de emisores de confianza.

- `schemeInformation`: Información asociada al esquema de la lista blanca de emisores de confianza.
    - `tislVersionIdentifier`: Versión del esquema de la lista blanca de emisores de confianza.
    - `schemeName`: Nombre, es los diferentes idiomas, del esquema de la lista blanca de emisores de confianza.
        - `lang: Idioma`
        - `text: Nombre del esquema`
- `trustIssuerList`: Listado de emisores de confianza.
    - `issuerName`: Nombre del emisor en diferentes idiomas.
        - `lang: Idioma`
        - `text: Nombre del emisor`
    - `issuerAddress`: Dirección del emisor.
    - `postalAddress`: Direcciones postales del emisor en diferentes idiomas.
        - `lang`: Idioma
        - `streetAddress`: Dirección de la calle.
        - `locality`: Ciudad o localidad.
        - `stateOrProvince`: Estado o provincia.
        - `postalCode`: Código postal.
        - `countryName`: Nombre del país.
    - `electronicAddress`: Dirección electrónica del emisor en diferentes idiomas.
        - `lang`: Idioma.
        - `text`: Dirección electrónica del emisor
    - `schemeTerritory`: Territorio al que pertenece el esquema del emisor.
    - `authorizedToIssue`: Listado de credenciales que el emisor está autorizado a emitir.
    - `policyOrLegalNotice`: Avisos legales o políticas asociadas con el emisor.
        - `tislLegalNotice`: Aviso legal en un idioma especifico.
            - `Lang`: Idioma.
            - `Text`: Contenido del aviso legal.
    - `serviceDigitalIdentities`: Identidades digitales de los servicios proporcionados por el emisor.
        - `digitalId`: Identificador digital del servicio.
            - `did`: Identificador descentralizado del emisor, generado a partir de la clave pública.
            - `x509Certificate`: Certificado X.509 asociado a la identidad digital del servicio.

2.6.2 Lista blanca de proveedores de contenido

A continuación, se muestra un modelo con todas las etiquetas habilitadas de la lista blanca en la que los proveedores de contenido de confianza deberán estar registrados antes de poder solicitar la presentación de credenciales:


```
{
  "trustContentProviderStatusList": {
    "id": "TCPSL20240326",
    "nextUpdate": {
      "dateTime": "2024-09-22T00:00:00Z"
    },
    "distributionPoints": {
      "uri": [
        "URI de publicación de la lista blanca"
      ]
    },
    "schemeInformation": {
      "tcpslVersionIdentifier": "5",
      "schemeName": [
        {
          "lang": "en",
          "text": "EN:Trusted list including information related to the content providers"
        },
        {
          "lang": "es",
          "text": "ES:Lista de confianza que incluye informacion relacionada con los proveedores de contenido"
        }
      ]
    },
    "trustContentProviderList": [
      {
        "contentProviderName": [
          {
            "lang": "es",
            "text": "Todo Porno España"
          },
          {
            "lang": "en",
            "text": "Todo Porno España"
          }
        ],
        "clientUri": "https://wwww.todoporno.es/postpresvc",
        "responseUri": "https://wwww.todoporno.es/postpresvc",
        "requestUri": "https://wwww.todoporno.es/request.json?id=001",
        "contentProviderAddress": {
          "postalAddress": [
            {
              "lang": "en",
              "streetAddress": "Registered address of the content provider",
              "locality": "Madrid",
              "stateOrProvince": "Madrid",
              "postalCode": "28020",
              "countryName": "ES"
            },
            {
              "lang": "es",
              "streetAddress": "Domicilio social del proveedor de contenido",
              "locality": "Madrid",
              "stateOrProvince": "Madrid",
              "postalCode": "28020",
              "countryName": "ES"
            }
          ],
```

```
          "electronicAddress": [
            {
              "lang": "es",
              "text": "https://wwww.todoporno.es"
            },
            {
              "lang": "es",
              "text": "mailto:todoporno@todoporno.es"
            }
          ]
        },
        "schemeTerritory": "ES",
        "authorizedToRequest": [
          "K",
          "UD"
        ],
        "policyOrLegalNotice": {
          "tcpslLegalNotice": [
            {
              "lang": "en",
              "text": "The applicable legal framework for the present trusted list"
            },
            {
              "lang": "es",
              "text": "El marco juridico aplicable a la presente lista de confianza"
            }
          ]
        },
        "serviceDigitalIdentities": [
          {
            "clientId": "https://wwww.todoporno.es/request.json?id=001"
          }
        ]
      }
    ]
  }
}
```


Figura 8. Lista blanca de proveedores de contenido restringido para adultos

- `trustContentProviderStatusList`: Elemento raíz que contiene toda la lista de proveedores de contenido de confianza.
    - id: Identificador único de la lista de proveedores de contenido de confianza.
- `nextUpdate`: Fecha y hora en la que se espera que la lista de proveedores de contenido de confianza se actualice.
- `distributionPoints`: Puntos de distribución donde se puede obtener la lista blanca de proveedores de contenido de confianza.
    - uri: Dirección URL donde se puede obtener la lista blanca de proveedores de contenido de confianza.
- `schemeInformation`: Información asociada al esquema de la lista blanca de proveedores de contenido de confianza.
    - `tcpslVersionIdentifier`: Versión del esquema de la lista blanca de proveedores de contenido de confianza.
    - `schemeName`: Nombre del esquema de la lista blanca de proveedores de contenido de confianza.
        - `lang: Idioma`
        - `text: Nombre del esquema`
- `trustContentProviderList`: Listado de proveedores de contenido de confianza.

- `contentProviderName`: Nombre del proveedor de contenido en diferentes idiomas.
  - `lang: Idioma`
  - `text: Nombre del proveedor de contenido`
- `clientURI`: URI identificativa del proveedor de contenido.
- `responseURI`: URI de respuesta del proveedor de contenido.
- `requestURI`: URI de solicitud del proveedor de contenido.
- `contentProviderAddress`: Dirección del proveedor de contenido.
  - `postalAddresses`: Contenedor para las direcciones postales del proveedor de contenido en diferentes idiomas.
    - `streetAddress`: Dirección de la calle.
    - `locality`: Ciudad o localidad.
    - `stateOrProvince`: Estado o provincia.
    - `postalCode`: Código postal.
    - `countryName`: Nombre del país.
    - `lang`: Idioma
  - `electronicAddress`: Dirección electrónica del proveedor de contenido en diferentes idiomas.
    - lang: Idioma
    - lext: Dirección electrónica del proveedor
- `schemeTerritory`: Territorio al que pertenece el esquema del proveedor de contenido.
- `authorizedToRequest`: Listado de credenciales que el emisor está autorizado a solicitar.
- `policyOrLegalNotice`: Avisos legales o políticas asociadas con el proveedor de contenido.
  - `tcpslLegalNotice`: Aviso legal en un idioma especifico.
    - `lang`: Idioma.
    - `text`: Contenido del aviso legal.
- `serviceDigitalIdentities`: Identidades digitales de los servicios proporcionados por el proveedor de contenido.
  - `serviceDigitalIdentity`: Identidad digital de un servicio proporcionado por el proveedor de contenido.
    - `clientId`: Identificador del cliente del servicio proporcionado por el proveedor de contenido.

3 ACUERDO DE INTERFACES

La siguiente tabla describe el acuerdo de interfaces del protocolo de presentación de la evidencia:

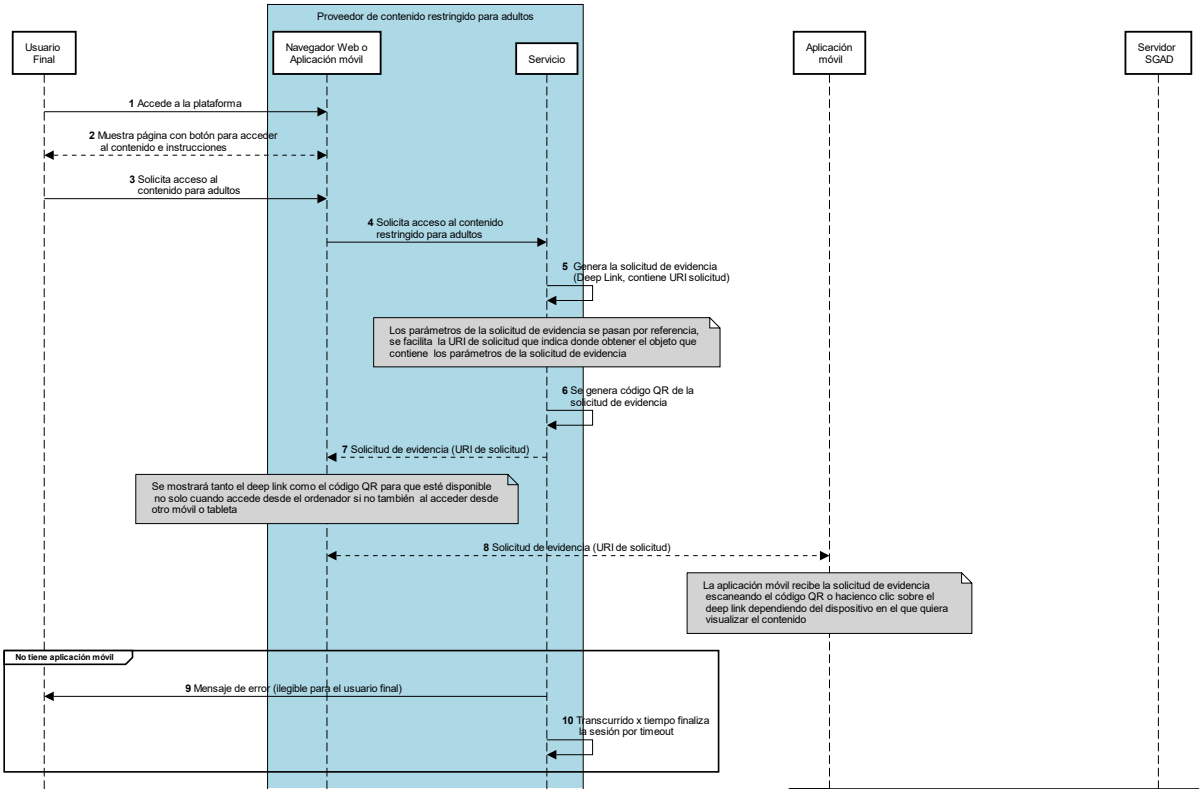
Servicio	Tipo de servicio	Esquema URL	Datos de entrada		Datos de salida
			Formato	Query Parameters/Body	
Solicitud de evidencia	GET	ageverification://authorize	application/x-www-form-urlencoded	**Query Parameters:** • *client_id:* URL de respuesta a la que se enviará la evidencia • *request_uri:* URL que referencia los parámetros de solicitud de evidencia	No aplica
Solicitud del objeto de la evidencia	HTTPS GET	*request_uri* *Se obtiene en la solicitud de evidencia*	No aplica	No aplica	JSON. Objeto de la solicitud de la evidencia *véase modelo de datos*
Envío de evidencia	HTTPS POST	*client_id* *Se obtiene el client_id en el objeto de la solicitud de evidencia y debe de coincidir con el campo response_uri de este y con el campo client_id de la solicitud de evidencia*	application/x-www-form-urlencoded	**Body:** • response: JWT. Evidencia *Véase sección 2.3, modelo de datos de la evidencia* Ejemplo: response = eyJra...9thuie	• Petición procesada: HTTPS status code 200 • Petición mal realizada: HTTPS status code 400 • Error interno: HTTPS status code 500

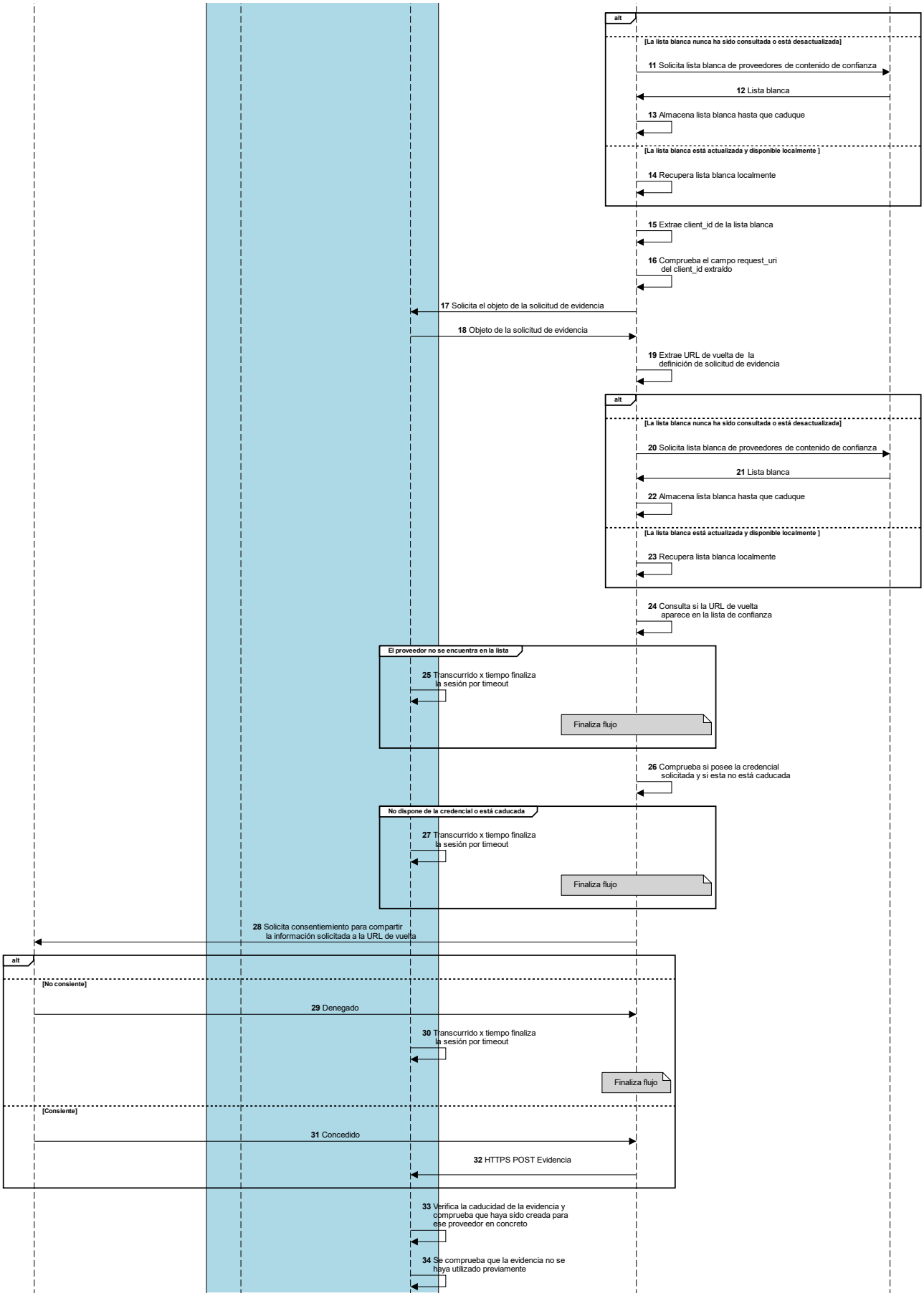
Tabla 1. Acuerdo de interfaces

La solicitud de evidencia es una redirección a la aplicación móvil que facilita los campos `client_id` y `request_uri` descritos en el modelo de datos previo. Mediante una petición HTTPS GET a la URI facilitada en el campo `request_uri` se obtiene como respuesta a la petición el objeto de la solicitud de evidencia. El último servicio es el del envío de la evidencia al cual se enviará la evidencia en formato JWT dentro del campo response en el body de la petición utilizando la codificación *application/x-www-form-urlencoded* mediante una petición HTTPS POST a la URL indicada en el campo *client_id* del objeto de solicitud obtenido previamente, nótese que, el campo `client_id` debe coincidir con el campo `response_uri` de dicho objeto y con el campo `client_id` de la solicitud de evidencia, y que la aplicación móvil debe verificar previamente que el proveedor de contenidos con identificador `client_id` se encuentre en la lista blanca de proveedores de contenido.

4 FLUJO DE ACCESO A CONTENIDO RESTRINGIDO PARA ADULTOS

Cuando el usuario final quiere acceder al contenido restringido para adultos, bien desde un navegador web o bien desde una aplicación móvil del proveedor de contenido, comienza el flujo de comunicación entre el proveedor de contenidos y la aplicación móvil Cartera Digital^{BETA} realizado mediante el protocolo OpenID For Verifiable Presentations [OpenID4VP] tal y como se muestra en la [Figura-OpenID4VP].


Proveedor de contenido restringido para adultos
Usuario Final
Navegador Web o Aplicación móvil
Servicio
Aplicación móvil
Servidor SGAD
1 Accede a la plataforma
2 Muestra página con botón para acceder al contenido e instrucciones
3 Solicita acceso al contenido para adultos
4 Solicita acceso al contenido restringido para adultos
5 Genera la solicitud de evidencia (Deep Link, contiene URI solicitud)
Los parámetros de la solicitud de evidencia se pasan por referencia, se facilita la URI de solicitud que indica donde obtener el objeto que contiene los parámetros de la solicitud de evidencia
6 Se genera código QR de la solicitud de evidencia
7 Solicitud de evidencia (URI de solicitud)
Se mostrará tanto el deep link como el código QR para que esté disponible no solo cuando accede desde el ordenador si no también al acceder desde otro móvil o tableta
8 Solicitud de evidencia (URI de solicitud)
La aplicación móvil recibe la solicitud de evidencia escaneando el código QR o haciendo clic sobre el deep link dependiendo del dispositivo en el que quiera visualizar el contenido
No tiene aplicación móvil
9 Mensaje de error (ilegible para el usuario final)
10 Transcurrido x tiempo finaliza la sesión por timeout


alt
[La lista blanca nunca ha sido consultada o está desactualizada]
11 Solicita lista blanca de proveedores de contenido de confianza
12 Lista blanca
13 Almacena lista blanca hasta que caduque
[La lista blanca está actualizada y disponible localmente]
14 Recupera lista blanca localmente
15 Extrae client_id de la lista blanca
16 Comprueba el campo request_uri del client_id extraído
17 Solicita el objeto de la solicitud de evidencia
18 Objeto de la solicitud de evidencia
19 Extrae URL de vuelta de la definición de solicitud de evidencia
alt
[La lista blanca nunca ha sido consultada o está desactualizada]
20 Solicita lista blanca de proveedores de contenido de confianza
21 Lista blanca
22 Almacena lista blanca hasta que caduque
[La lista blanca está actualizada y disponible localmente]
23 Recupera lista blanca localmente
24 Consulta si la URL de vuelta aparece en la lista de confianza
El proveedor no se encuentra en la lista
25 Transcurrido x tiempo finaliza la sesión por timeout
Finaliza flujo
26 Comprueba si posee la credencial solicitada y si esta no está caducada
No dispone de la credencial o está caducada
27 Transcurrido x tiempo finaliza la sesión por timeout
Finaliza flujo
28 Solicita consentiemiento para compartir la información solicitada a la URL de vuelta
alt
[No consiente]
29 Denegado
30 Transcurrido x tiempo finaliza la sesión por timeout
Finaliza flujo
[Consiente]
31 Concedido
32 HTTPS POST Evidencia
33 Verifica la caducidad de la evidencia y comprueba que haya sido creada para ese proveedor en concreto
34 Se comprueba que la evidencia no se haya utilizado previamente

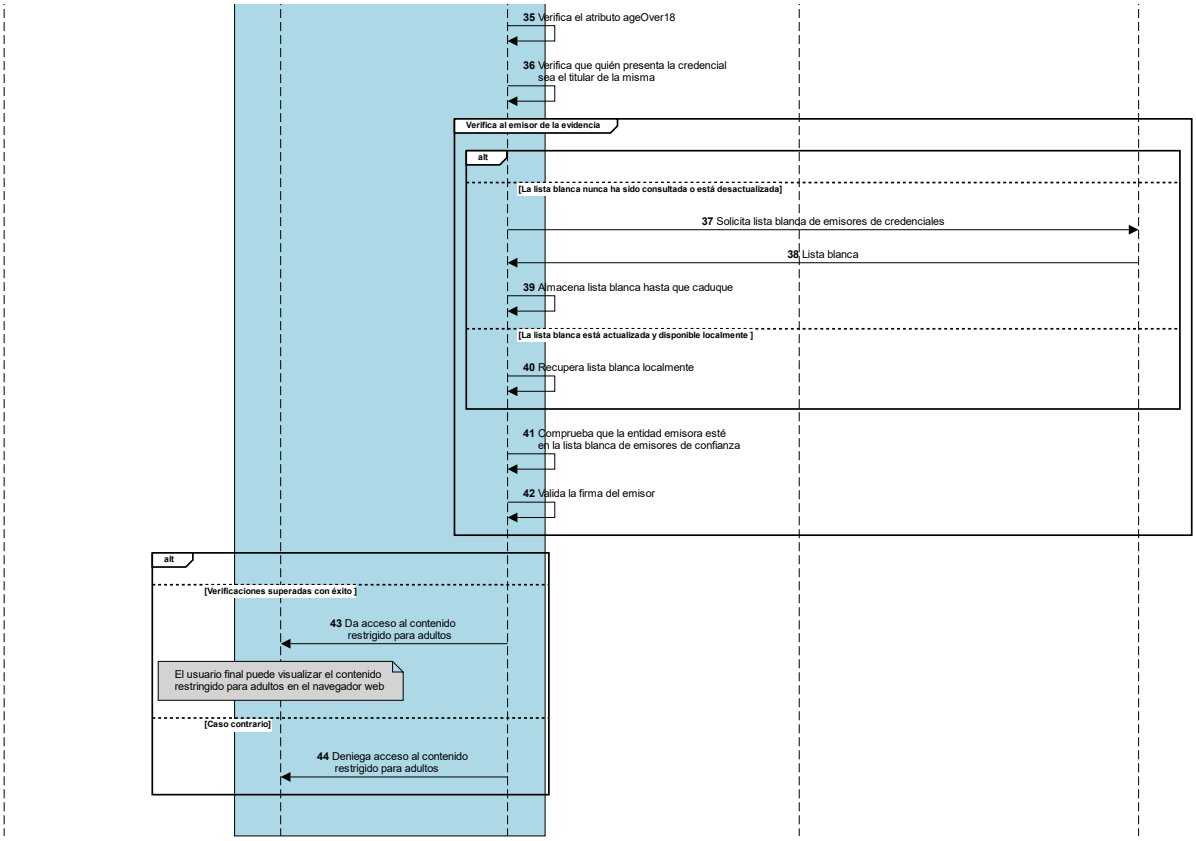


Figura 9. Flujo OID4VP

El flujo mediante el cual el proveedor de contenido verifica si el usuario final es mayor de edad para permitir el acceso al contenido restringido para adultos está conformado por los siguientes pasos:

1. El usuario final solicita el acceso al contenido restringido para adultos desde la plataforma del proveedor de contenidos.
2. La plataforma muestra un botón para acceder al contenido junto a las instrucciones de los siguientes prerrequisitos:
   - Tener la aplicación móvil Cartera Digital [BETA] instalada junto al enlace indicando donde se puede descargar.
   - Tener la credencial de mayoría de edad.
3. El usuario final hace clic sobre el botón para acceder al contenido restringido para adultos.
4. La plataforma solicita al servicio acceso al contenido.
5. El servicio del proveedor de contenidos genera la solicitud de evidencia.
6. Se genera un código QR a partir de la solicitud de evidencia, véase ISO/IEC 18004:2015.
7. El servicio del proveedor de contenidos devuelve a la plataforma correspondiente la solicitud de evidencia.

El proveedor de contenido mostrará en la plataforma tanto el código QR como el *Deep Link* de la solicitud de evidencia de forma que, el usuario final pueda acceder con la aplicación móvil Cartera Digital[BETA] independientemente de si está instalada en el mismo dispositivo o en otro diferente a aquel en el que quiere visualizar el contenido.

8. La aplicación móvil Cartera Digital[BETA] obtiene la solicitud de evidencia, bien escaneando el código QR o bien haciendo clic sobre la URI de la solicitud de evidencia que redireccionará al usuario final a la aplicación móvil.

En caso de que el usuario final no tenga la aplicación Cartera Digital [BETA] instalada en el dispositivo móvil:

9. El proveedor de contenido utilizará el campo `nonce` de la solicitud de evidencia para controlar el *timeout* de la sesión, se establecerá a 2 minutos, si tras dos minutos no recibe la evidencia, como sucederá en el caso en el que el usuario no tiene descargada la aplicación móvil, el proveedor de contenidos cerrará la sesión.

La aplicación móvil podrá almacenar la lista blanca en caché hasta que la propiedad `nextUpdate` supere la fecha en la que desea consultar la lista. Por tanto, si no dispone localmente de la lista blanca de proveedores o la fecha es superior a la indicada en el campo `nextUpdate` de la lista:

10. La aplicación móvil solicita la lista blanca de proveedores de contenido.
11. Obtiene la lista blanca.
12. Almacena la lista localmente.

En caso de que disponga de la lista y la fecha en la que consulta esta sea inferior a la indicada en el campo `nextUpdate`:

13. Obtiene la lista blanca.
14. Utiliza el campo `client_id` que ha recibido en la solicitud de evidencia para obtener la información del proveedor de contenidos asociado a ese identificador de la lista blanca.
15. Comprueba si el proveedor de contenido asociado a ese identificador tiene en la lista blanca de proveedores de contenido asociada la `request_uri` que se indica en la solicitud de evidencia.
16. Si es de confianza, es decir, `si` en la lista blanca existe el `client_id` y tiene asociado el `request_uri` facilitado en la solicitud, la aplicación móvil solicita el objeto que contiene los parámetros de la solicitud de evidencia realizando una petición GET a la URI que se ha proporcionado en el campo `request_uri` de la solicitud de evidencia

```
GET /request.jwt/GkurKxf5T0Y-mnPFCHqWOMiZi4VS138cQO_V7PZHAdM
```

    recibida previamente.
17. Obtiene el objeto con los parámetros de la solicitud de evidencia.
18. La aplicación móvil extrae el parámetro `response_uri` del objeto obtenido en el paso previo.

En caso de que la aplicación móvil no disponga localmente de la lista blanca de proveedores o la fecha sea superior a la indicada en el campo `nextUpdate` de la lista:

19. Solicita al servidor público la lista blanca de proveedores de contenido de confianza.
20. Obtiene la lista blanca.
21. Almacena la lista localmente.

En caso de que disponga de la lista y la fecha en la que consulta esta sea inferior a la indicada en el campo `nextUpdate`:

22. Recupera la lista blanca localmente.
23. Consulta en la lista blanca si el valor `response_uri` que viene en el objeto que de la solicitud de evidencia se encuentra en la lista blanca, puesto que es la URI de respuesta la que se utiliza en la lista blanca como identificador del proveedor de confianza.

En caso de que el proveedor no estuviese en la lista blanca, no sería una entidad de confianza y no se le daría ninguna respuesta, de forma que,

24. Transcurridos los dos minutos de *timeout* el proveedor cerrará la sesión.

Si por el contrario el proveedor es una entidad de confianza,

25. La aplicación móvil comprobará si posee la credencial que le ha sido solicitada.

En caso de no disponer de la credencial, el proveedor no recibirá respuesta alguna para que no pueda deducir información sobre el usuario final, transcurridos los dos minutos configurados en el proveedor para el timeout,

26. El proveedor cerrará la sesión.

En caso contrario,

27. Se le solicitará consentimiento al usuario final para compartir la credencial de mayoría de edad con el proveedor de contenidos.

Si el usuario no quiere compartir la credencial,

28. Deniega el consentimiento.
29. Transcurridos dos minutos el proveedor cerrará la sesión.

Si el usuario quiere compartir la credencial,

30. Da su consentimiento para compartir la credencial.
31. La aplicación móvil realiza una petición POST al servicio del proveedor con la evidencia.

Una vez que el proveedor obtiene la evidencia comienza el proceso de verificación detallado en la siguiente sección. Finalmente, si se superan todas las validaciones con éxito, el servicio del proveedor dará acceso al contenido restringido para adultos a la plataforma donde el usuario final lo podrá visualizar, en caso contrario, el usuario no podrá acceder al contenido.

5 VERIFICACIÓN DE LA CREDENCIAL DE MAYORÍA DE EDAD

Una vez que el proveedor de identidad ha obtenido la presentación verificable, deberá realizar las siguientes validaciones sobre la misma para dar acceso al usuario final al contenido restringido para adultos:

1. A partir del campo `nonce` recibido en la evidencia se recuperará localmente tanto la sesión como la solicitud de autorización asociada. Se deberá comprobar que no haya sido previamente utilizado.

2. Se deberá verificar que tanto la evidencia como las presentaciones verificables incluidas en su interior no estén caducadas, evaluando el valor dado por el campo `exp` incluida en los distintos tokens JWT. De igual forma, se deberá verificar que estas hayan sido generadas para el proveedor de contenido esperado, evaluando en este caso que el campo `aud` coincida con el identificador único del proveedor.
3. Se deberá verificar que tanto la evidencia como la presentación verificable incluida en ella haya sido firmada por el propietario de la credencial que se presenta en la misma, es decir, por la clave privada asociada al DID que se encuentre en el campo *credentialSubject.id* de la credencial.
4. Se deberá verificar que cada una de las presentaciones y credenciales incluidas en la evidencia cumplen con los requerimientos que se indicaron en el campo `presentation_definition` de la solicitud de autorización. Tomando como referencia el modelo de solicitud de respuesta definido en este documento, se deberán realizar las siguientes verificaciones:
    - En la evidencia deben venir incluidas todas las presentaciones verificables requeridas. Concretamente, para este caso de uso, se debe asegurar la recepción de una única presentación verificable en formato JWT. Para ello se debe consultar el campo `vp_token` de la evidencia.
    - En el conjunto de presentaciones verificables incluidas en la evidencia, deben venir incluidas todas las credenciales verificables requeridas, así como la ubicación de cada una de estas. En otros términos, para cada elemento (id) del campo *input_descriptor* de la solicitud de autorización deberá existir un elemento asociado (id) en el campo *vp_token.presentation_submission.descriptor_map* de la evidencia. Para este caso de uso se espera una única presentación con la credencial de mayoría de edad.
    - El campo *presentation_definition.id* debe coincidir con el campo *presentation_submission.definition_id* de la evidencia
    - La presentación debe venir en formato JWT y el algoritmo de la firma, dado por el campo presentation_definition.format, debe ser RS512
    - Para cada credencial ‘j’, de una presentación dada ‘i’, requerida por el campo *presentation_definition[i].input_descriptors[j].id* en la solicitud de autorización, debe existir una credencial análoga en el campo presentation_submission[i].descriptor_map[j].id
    - Para cada restricción `path` definida para una credencial ‘j’, de una presentación dada ‘i’, se debe validar la existencia de ese campo en la credencial y ubicación especificadas. Para ello se debe consultar lo definido en el campo por el campo *presentation_definition[i].input_descriptors[j].constraints.fields.path.*
5. Se deberá verificar que ninguna de las credenciales verificables requeridas se encuentre caducada. Para ello se debe consultar, si existe, el campo *validUntil* de la credencial.
6. Se deberá verificar que el valor del campo tipo de credencial es K ("type": ["VerifiableCredential", “K"]), este atributo es el que acredita que el titular de la credencial puede acceder al contenido solicitado.

7. El verificador deberá extraer la clave pública del DID del emisor y validar la firma en cada una de las credenciales haciendo uso de esa clave. A continuación, el verificador deberá comprobar que el DID asociado al emisor se encuentra en la lista blanca de emisores. Para ello, este debe mantener un registro local actualizado de dicha lista. Se usará preferentemente la lista local para realizar las comprobaciones, a menos que esta no haya sido descargada aún o se haya superado el tiempo máximo para el que la lista se encuentra actualizada, en cuyo caso el verificador tendrá que descargarla del repositorio habilitado para ello.

6 ANEXO I – REFERENCIAS

[DID-Core] Sporny, M., Guy, A., Sabadello, M., and D. Reed, "Decentralized Identifiers (DIDs) v1.0", 19 July 2022, <https://www.w3.org/TR/did-core/>.

[DID-Key] Sporny, M., Zagidulin D., Longley D., Steele O., "The did:key Method v0.7", 02 September 2022, <https://w3c-ccg.github.io/did-method-key/>.

[OpenID4VP] Terbu, O., Lodderstedt, T., Yasuda, K., and T. Looker, "OpenID for Verifiable Presentations", 29 November 2022, <https://openid.net/specs/openid-4-verifiable-presentations-1_0.html>.

[DIF.PresentationExchange] Buchner, D., Zundel, B., Riedel, M., and K. H. Duffy, "Presentation Exchange 2.0.0", <https://identity.foundation/presentation-exchange/spec/v2.0.0/>.

[DIF.ClaimFormatRegistry] Buchner, D., Zundel, B., Riedel, M., and K. H. Duffy, "Claim Format Registry", <https://identity.foundation/claim-format-registry/#registry>.

[RFC2397] L. Masinter, "The "data" URL scheme", <https://www.rfc-editor.org/rfc/rfc2397>.

[RFC7515] Jones, M., Bradley, J., and N. Sakimura, "JSON Web Signature (JWS)", RFC 7515, DOI 10.17487/RFC7515, May 2015, <https://datatracker.ietf.org/doc/html/rfc7515>.

[RFC7518] Jones, M., "JSON Web Algorithms (JWA)", RFC 7518, DOI 10.17487/RFC7518, May 2015, <https://datatracker.ietf.org/doc/html/rfc7518 >.

[Figura-OpenID4VP] Anexo II – OpenID4VP.svg