Age verification system for access to online content

Age verification ecosystem

Version 1

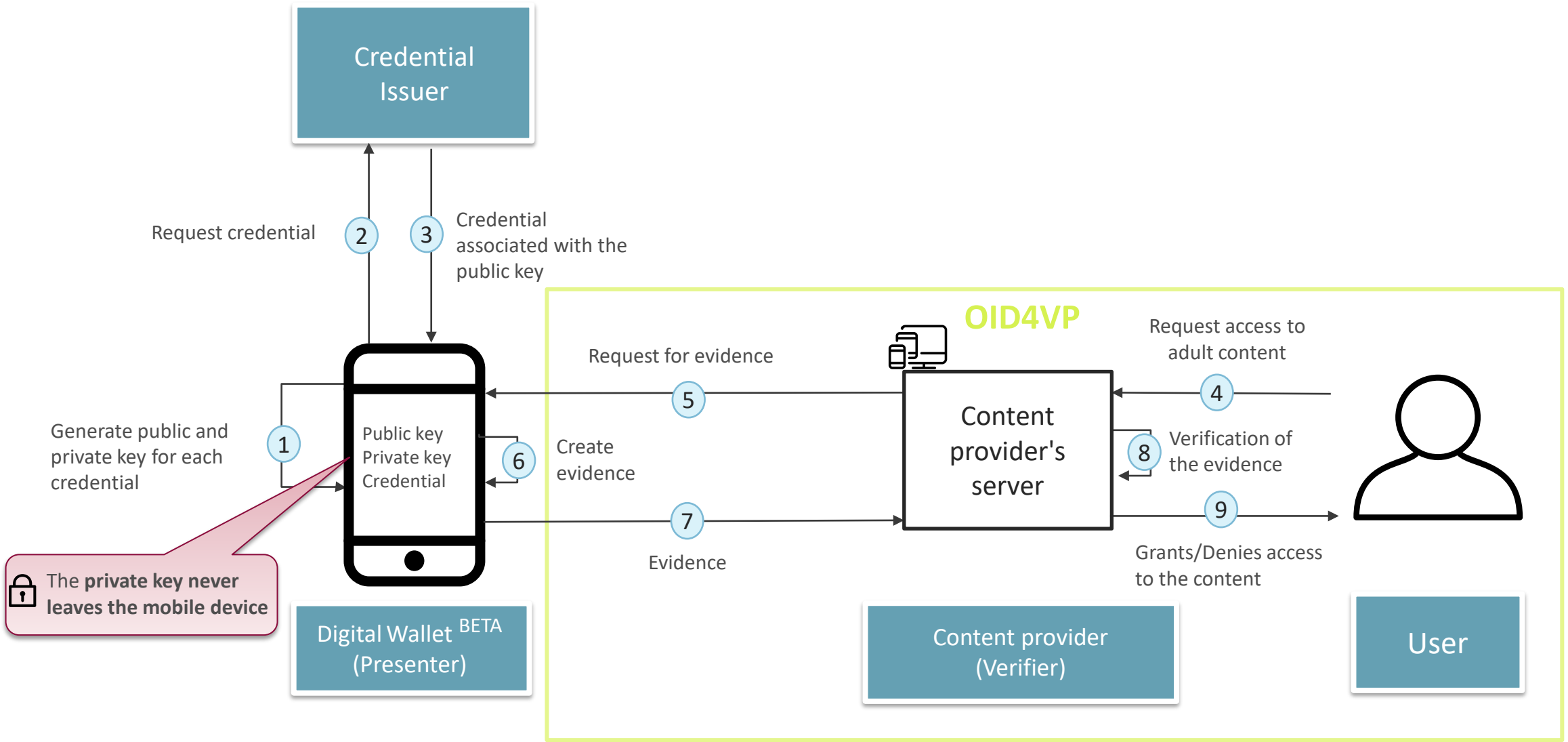
June 30, 2024

GOBIERNO DE ESPAÑA
MINISTERIO PARA LA TRANSFORMACIÓN DIGITAL Y DE LA FUNCIÓN PÚBLICA
SECRETARÍA GENERAL DE ADMINISTRACIÓN DIGITAL
Plan de Recuperación, Transformación y Resiliencia
Financiado por la Unión Europea NextGenerationEU
España | digital 2026

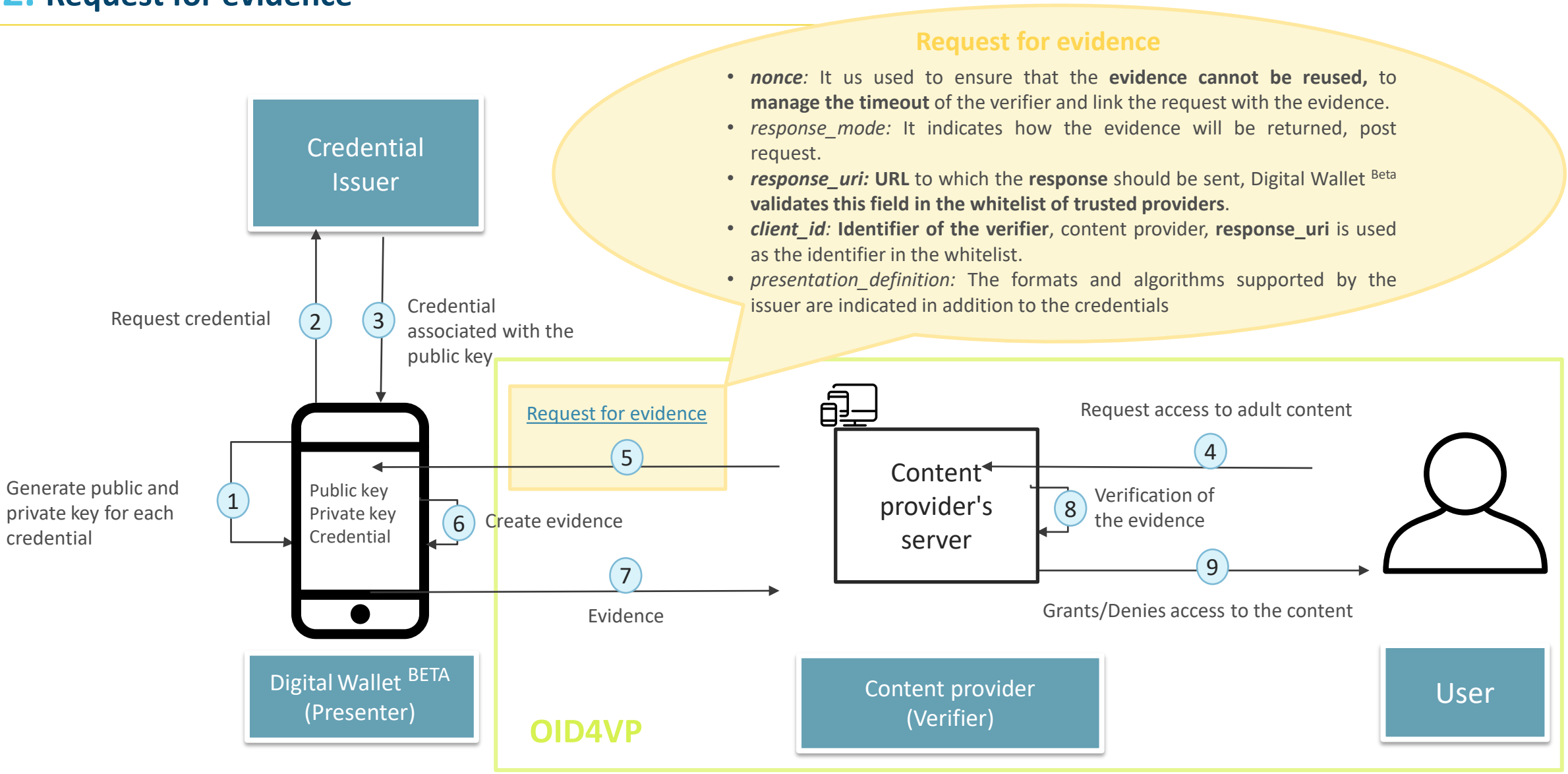
ÍNDICE

Solution components

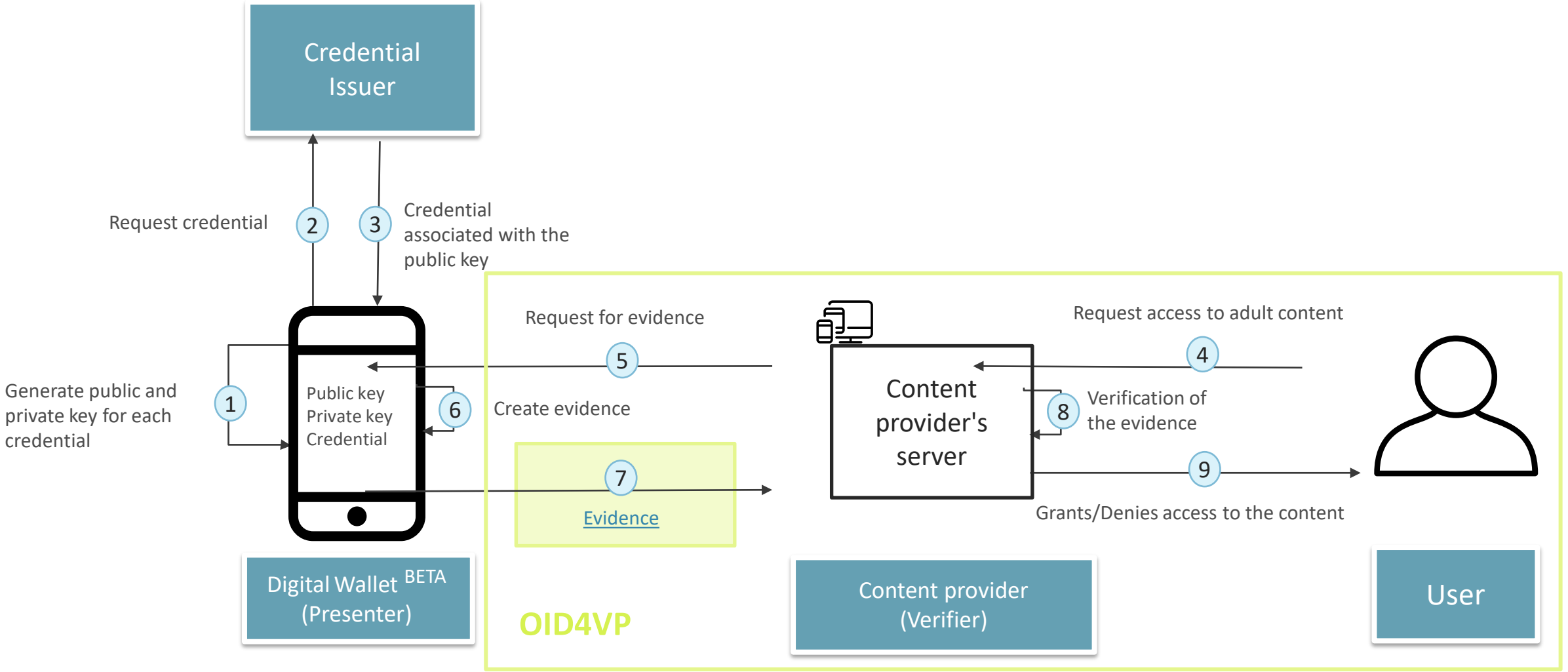
1. General solution components



2. Request for evidence



3. Evidence


Credential Issuer
Request credential
2
3
Credential associated with the public key
Generate public and private key for each credential
1
Public key
Private key
Credential
6
Request for evidence
5
Create evidence
7
Evidence
Digital Wallet BETA (Presenter)
OID4VP
Content provider's server
Request access to adult content
4
8
Verification of the evidence
9
Grants/Denies access to the content
Content provider (Verifier)
User

4. Evidence

Evidence (OID4VP)

Header

(*typ, cty…*)

Body

(*iat, exp, aud…*)
presentation_submission (Used to indicate to the verifier formats, algorithms,...)
***nonce**: "*1b0a82db-9c82-4693-b9ca-97f62f4d5081"

Verifiable presentation (W3C)

Header

(*typ, cty…*)

Body

iat (Creation date)
exp (**Evidence expiration date**, could be **1 minute**)
aud (**Entity for which the evidence is generated***)*

Verifiable Credential (W3C)

Header

(*typ, cty…*)

Body

type: ["VerifiableCredential", "K"] (Attribute indicating whether the user is authorized to access)
id: "did:key:${ClavePúblicaUsuario}"
validUntil: "2024-05-08T10:59:52Z" (Expiration date, expires in one month)
iss: "did:key:${ClavePúblicaEmisor}"

Proof

d2k4O3FytQJf83kLh-HsXuPvh6yeOlhJETg… (**ISSUER signature**)

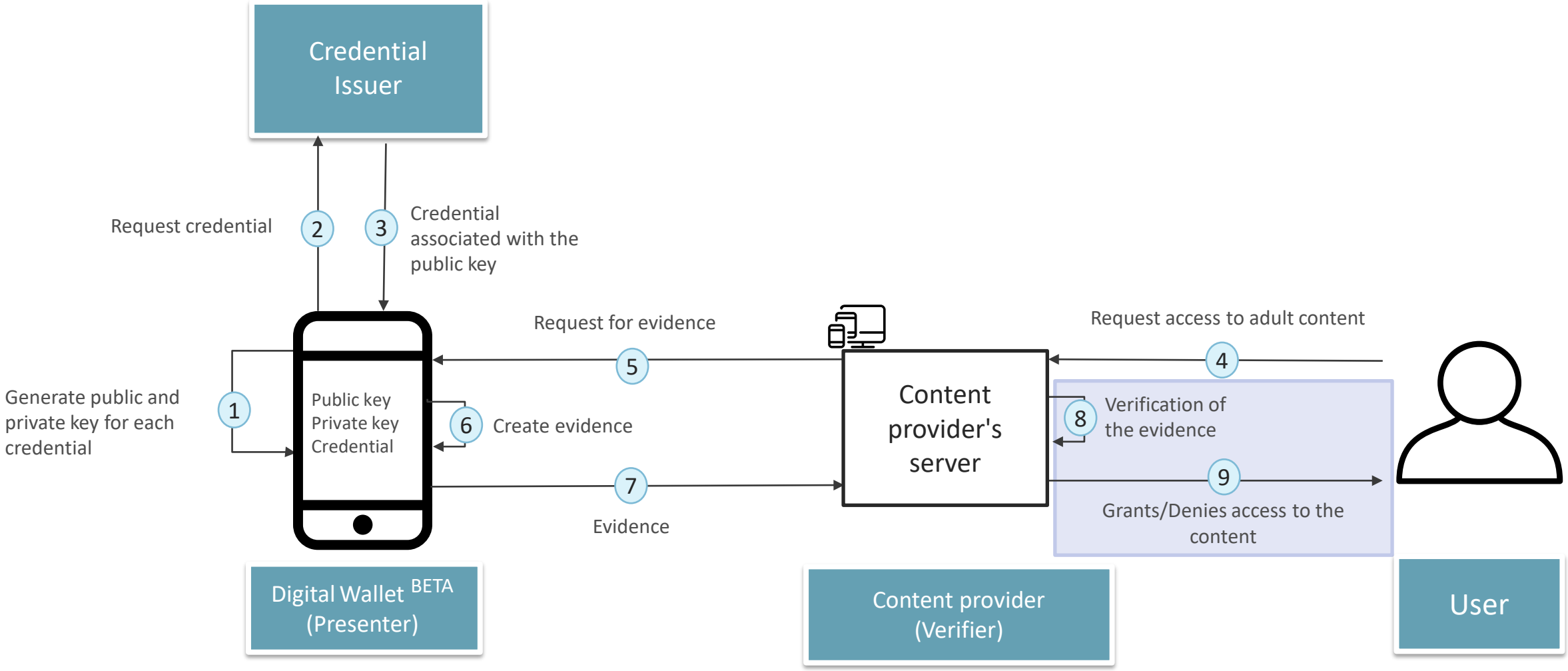
Proof

Xakou0923klmrw… (**USER signature**)

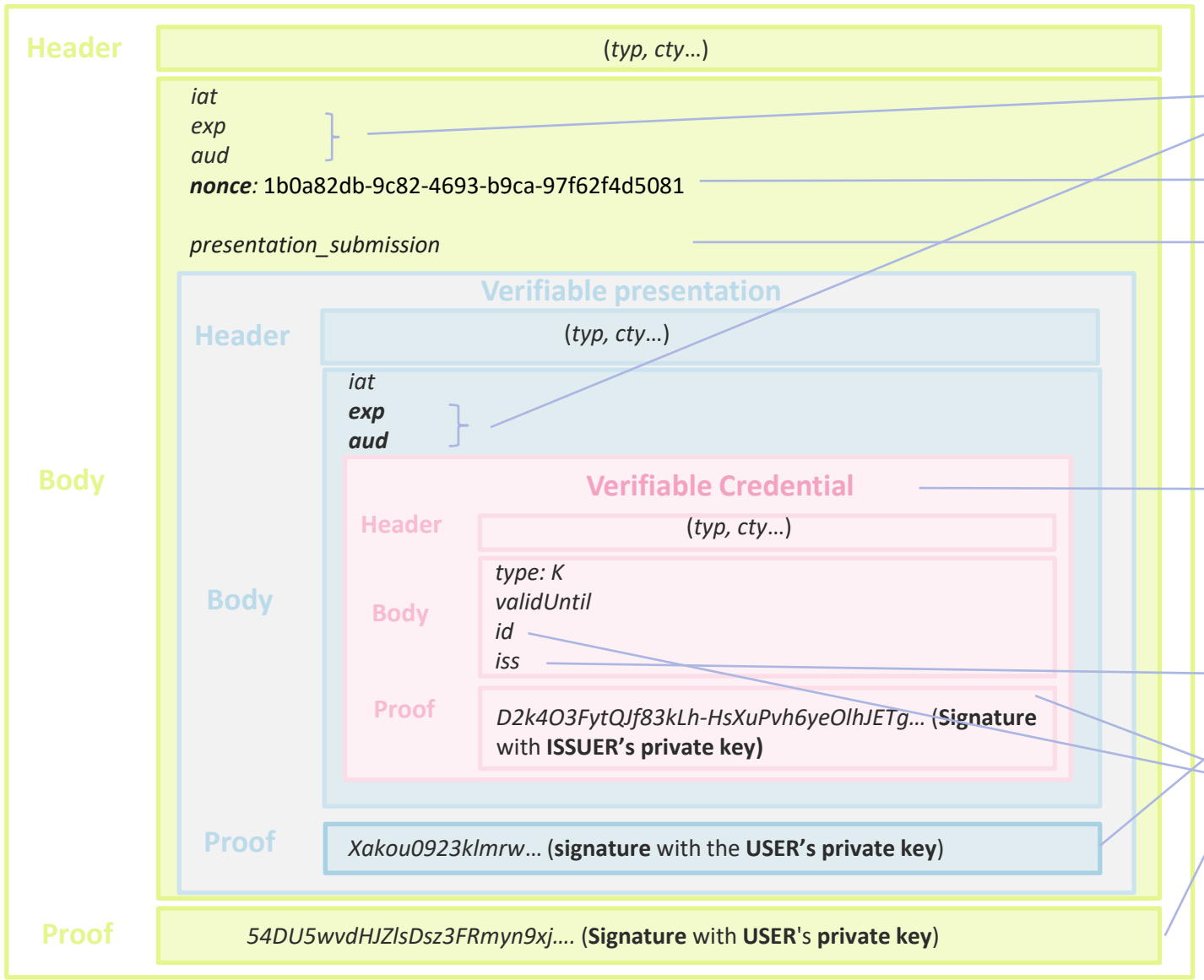
Proof

54DU5wvdHJZlsDsz3FRmyn9xj…. (**USER signature, so that other people's credentials cannot be presented**)

5. Verification of the evidence



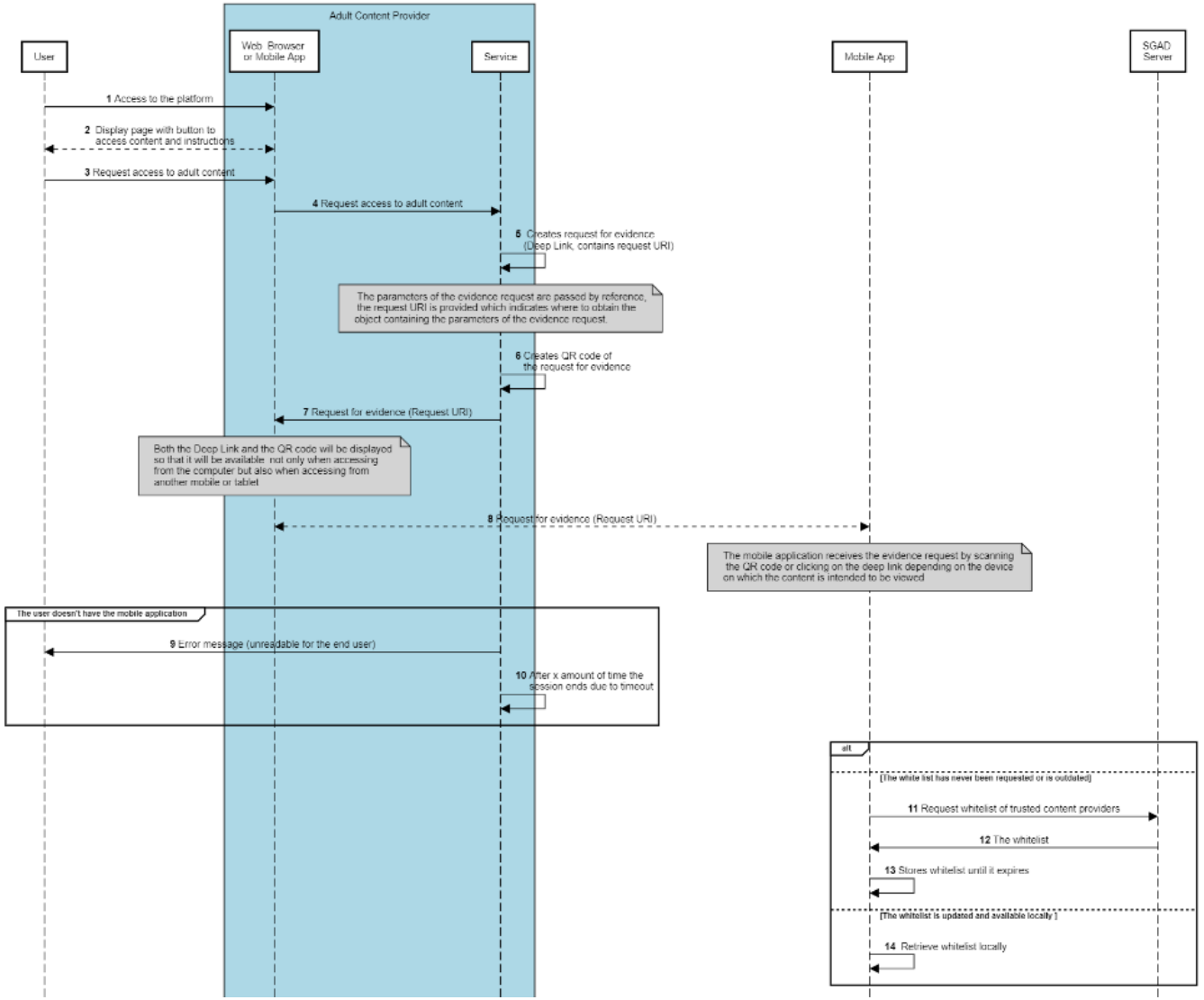
6. Verification of the evidence



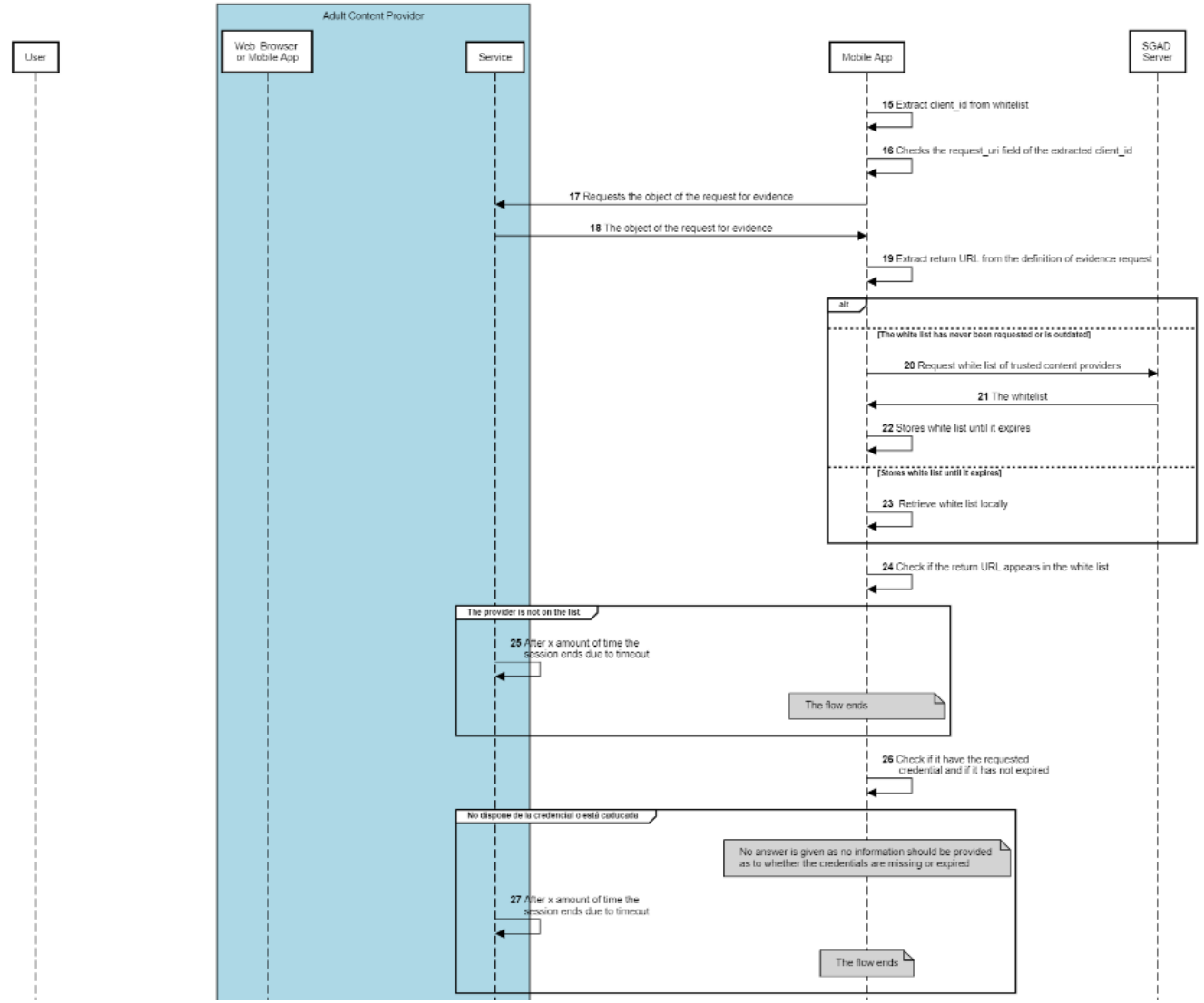
1. This is checked to ensure that it has not expired and that it has been generated for that specific content provider.
2. The session is recovered along with the request for evidence. It is checked to ensure that it has not been previously used.
3. This is verified to ensure that the request for evidence has been answered, for example, that the K credential is included *and* that formats and algorithms supported by the content provider are used.
4. It is checked to ensure it has not expired
5. It is checked that both signatures of the evidence coincide with the holder of the credential.
6. The credential issuer is checked to verify it is a trusted entity by consulting the whitelist of trusted issuers and the signature is validated with its public key.

Evidence presentation flow

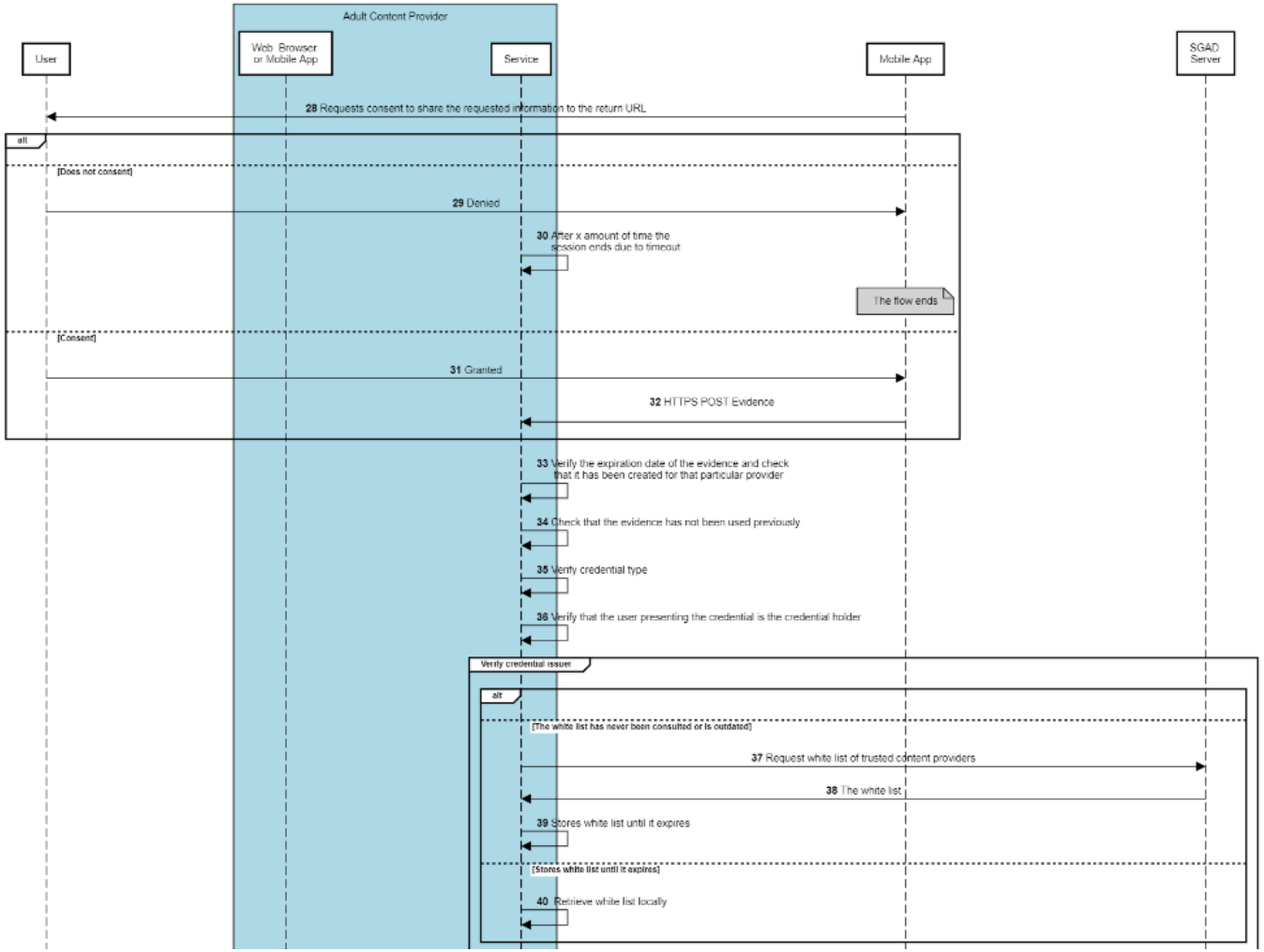
10. Evidence presentation flow



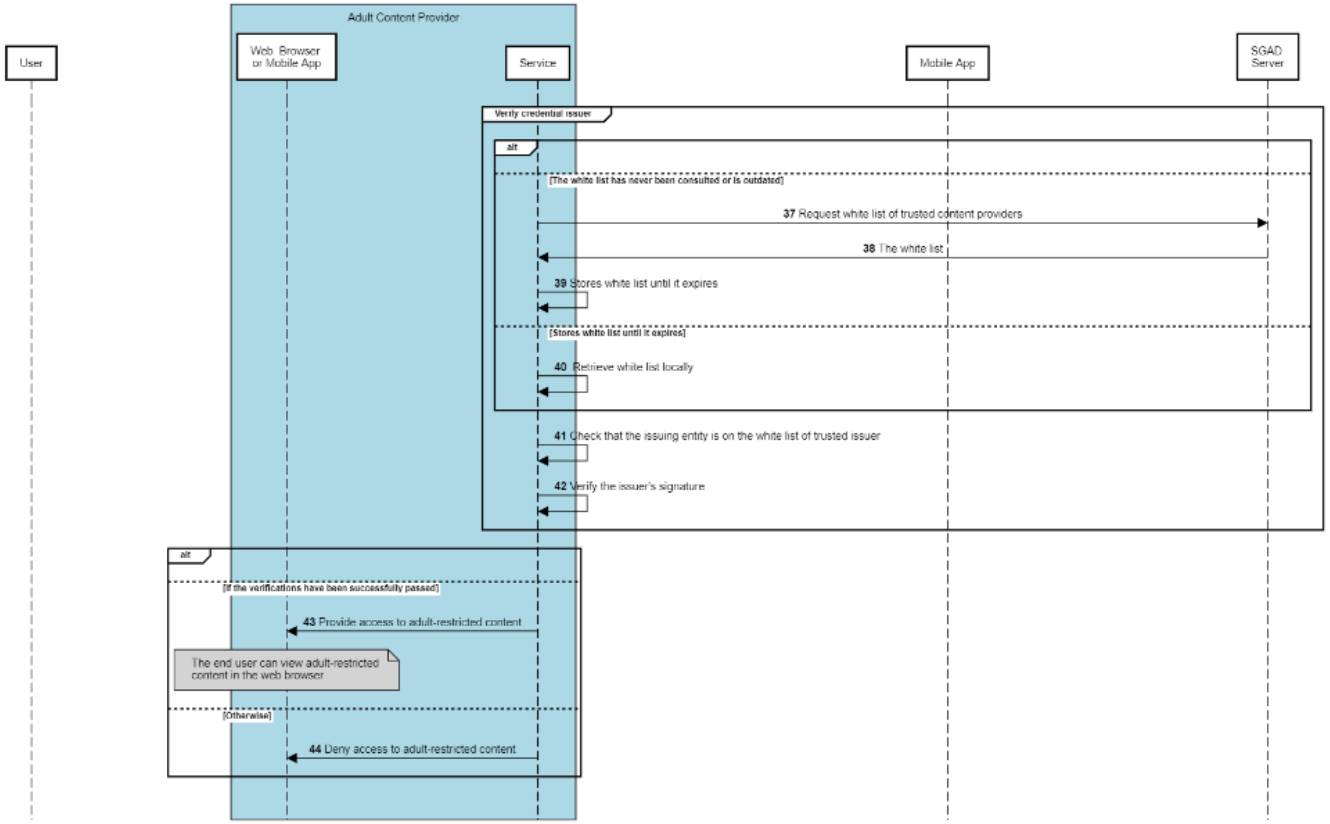
10. Evidence presentation flow



10. Evidence presentation flow



10. Evidence presentation flow



Solution data model

11. Data model - Verifiable presentation

Body of the Verifiable Presentation

```
{
  "id":"urn:uuid:00000000-0000-0000-0000-000000000000",# Unique identifier of the presentation
  "type": [
    "VerifiablePresentation "                          # Type of presentation
  ],
  "verifiableCredential":[                             # List of verifiable credentials included in the presentation
   {
    "@context" :"https://www.w3.org/ns/credentials/v2",  # Maps abbreviated concepts in credential to URLs
    "id":"data:application/vc+ld+json+jwt;${VCJWT}",     # Follows the RFC data URL, contains the verifiable credential in JWT format
    "type":"EnvelopedVerifiableCredential"               # Type stipulated in W3C for enveloped verifiable credentials
   }
  ],
  "holder":"did:key:z2dmzD81cgP…t35e "                 # Decentralized identifier generated from the user's public key
                                                       that generates the presentation. It must coincide with the holder of the
                                                       presented credentials
}
```

Go to the evidence

11. Data model - Verifiable presentation

Verifiable presentation secured with the end user's private key


```
eyJraWQiOiJFeEhrQk1XOWZtYmt2VjI2Nm1ScHVQMnNVWV9OX0VXSU4xbGFwVXpPOHJv
IiwiYWxnIjoiRVMzODQifQ.eyJAY29udGV4dCI6WyJodHRwczovL3d3dy53My5vcmcvb
nMvY3JlZGVudGlhbHMvdjIiLCJodHRwczovL3d3dy53My5vcmcvbnMvY3JlZGVudGlhb
HMvZXhhbXBsZXMvdjIiXSwidHlwZSI6IlZlcmlmaWFibGVQcmVzZW50YXRpb24iLCJ2Z
XJpZmlhYmxlQ3JlZGVudGlhbCI6W3siQGNvbnRleHQiOiJodHRwczovL3d3dy53My5vc
mcvbnMvY3JlZGVudGlhbHMvdjIiLCJpZCI6ImRhdGE6YXBwbGljYXRpb24vdmMrbGQra
nNvbitqd3Q7ZXlKcmFXUWlPaUpGZUVoclFrMVhPV1p0WW10MlZqSTJObTFTY0hWUU1uT
lZXVjlPWDBWWFNVNHhiR0Z3VlhwUE9ISnZJaXdpWVd4bklqb2lSVk16T0RRaWZRLmV5S
kFZMjl1ZEdWNGRDSTZXeUpvZEhSd2N6b3ZMM2QzZHk1M015NXZjbWN2Ym5NdlkzSmxaR
1Z1ZEdsaGJITXZkaklpTENKb2RIUndjem92TDNkM2R5NTNNeTV2Y21jdmJuTXZZM0psW
kdWdWRHbGhiSE12WlhoaGJYQnNaWE12ZGpJaVhTd2lhV1FpT2lKb2RIUndPaTh2ZFc1c
GRtVnljMmwwZVM1bGVHRnRjR3hsTDJOeVpXUmxiblJwWVd4ekx6RTROeklpTENKMGVYQ
mxJanBiSWxabGNtG1hV0ZpYkdWRGNtVmtaVzUwYVdGc0lpd2lSWGhoYlhCc1pVRnNkV
zF1YVVOeVpXUmxiblJwWVd3aVhTd2lhWE56ZFdWeUlqb2lhSFIwY0hNNkx5OTFibWwyW
lhKemFYUjVMbVY0WVcxd2JHVXZhWE56ZFdWeWN5ODFOalV3TkRraUxDSjJZV3hwWkVae
WIyMGlPaUl5TURFd0xUQXhMVEF4VkRFNU9qSXpPakkwV2lJc0ltTnlaV1JsYm5ScFlXe
FRZMmhsYldFaU9uc2lhV1FpT2lKb2RIUndjem92TDJWNFlXMXdiR1V1YjNKbkwyVjRZV
zF3YkdWekwyUmxaM0psWlM1cWMyOXVJaXdpZEhsd1pTSTZJa3B6YjI1VFkyaGxiV0VpZ
lN3aVkzSmxaR1Z1ZEdsaGJGTjFZbXBsWTNRaU9uc2lhV1FpT2lKa2FXUTZaWGhoYlhCc
1pUb3hNak1pTENKa1pXZHlaV1VpT25zaWRIbHdaU0k2SWtKaFkyaGxiRzl5UkdWbmNtV
mxJaXdpYm1GdFpTSTZJa0poWTJobGJHOXlJRzltSUZOamFXVnVZMlVnWVc1a0lFRnlkS
E1pZlgxOS5kMms0TzNGeXRRSmY4M2tMaC1Ic1h1UHZoNnllT2xoSkVMVm81VEY3MWd1N
2Vsc2xReU9mMlpJdEFYcnRiWEY0S3o5V2l2TmR6dE9heXo0VlVRME13YTh5Q0Raa1A5Q
jJwSC05U190Y0FGeGVvZUo2WjRYbkZ1TF9ET2ZrUjFmUCIsInR5cGUiOiJFbnZlbG9wZ
WRWZXJpZmlhYmxlQ3JlZGVudGlhbCJ9XX0.54DU5wvdHJZlsDsz3FRmyn9xj23IC56OW
6t6RQMQQuw9omLOxZ8Dkvg-12AADWJeKfYRaCKEIV7YmBkCe1JkQdV5NGxxtOvES4Ip-
VARZqVLi201sakDFERMMfbrBl7n
```


```
{
      iss: "did:key:z2dmzD81cgP…t35e“,    #evidence issuer
      “iat”: "did:key:z2dmzD81cgP…t35e“   #issuance date of the evidence
      “exp”: "1618496351",                #date evidence expires,
                                          format NumericDate JWT. Value
                                          set at 1 minute.
      “vp” :{
        "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;${VCJWT}",
            "type":"EnvelopedVerifiableCredential“
          }
        ],
       "holder":"did:key:z2dmzD81cgP…t35e“
      }
}
```

11. Data model - Verifiable presentation

Wrapped Verifiable Presentation (W3C)

```
{
  "@context": "https://www.w3.org/ns/credentials/v2",     #Maps abbreviated concepts in credential to URLs
  "id":"data:application/vp+ld+json+jwt;${VPJWT}”,        #Follows the RFC data URL, contains the verifiable credential in JWT
format
  "type": "EnvelopedVerifiablePresentation“               #Type stipulated in W3C for enveloped verifiable presentations
}
```

11. Data model- Request for evidence

Request for Evidence

URI referring to the data of the authorization request. It contains the following parameters in *application/x-www-form-urlencoded format:*

- request_uri: Absolute URI of the authorization object request. For example,

```
https%3A%2F%2Fauth.sitioadultos.com%2Frequest.json%2FGkurKxf5T0Y-mnPFCHqWOMiZi4VS138cQO_V7PZHAdM
```

- client_id: Content provider's identifier; the return URL set as the identifier in the whitelist will be used.

It is proposed that the authorization request be a *deep link*:

```
ageverification://authorize?client_id={response_uri}&request_uri=https%3A%2F%2Fauth.sitioadultos.com%2Frequest.json%2FGkurKxf5T0Y-mnPFCHqWOMiZi4VS138cQO_V7PZHAdM
```

10. Data model – Object of the request for evidence

Object of the request for evidence

```
{
        “response_type”: “vp_token”,                    #Indicates that the answer is a token, specifically, the token represents a verifiable presentation
        “client_id_schema”: “redirect_uri”,             #Client schema type, defines
        “response_mode”: “direct_post.jwt”,             #Response mode, since the verifier may be on a different device the authorization response
                                                        will be sent via POST request instead of redirection

        “response_uri”: ${URI de vuelta},               #URI to which the mobile application sends the authorization response. It must validate that it is trusted in the whitelist
                                                        of trusted providers where the identifier of each provider will be its return URI

        “client_id”: “${response_uri}”,                 #The return URI is used as client identifier, content provider
        “nonce”: “07d54d63-7136-3ff1-11d8-f 9d17bdb0620”,      #Unique identifier that the content provider uses to link the request with the response. It will be
                                                                used to manage the time that the session is kept open, and that the presentation is not reused

        “presentation_definition”: {
               “id”: “32f54163-7166-48f1-93d8-f f217bdb0653”,  #Unique identifier of the definition of the presentation
               “format”: {                                    #Formats supported by the verifier
                     “jwt_vc”: {
                         "alg": [“RS512”]                     #Algorithms supported by the verifier
                      },
                     “jwt_vp”: {
                         "alg": [“RS512”]                     #Algorithms supported by the verifier
                      },

               },
               “input_descriptors”: [{
                     “id”: “Age over 18”,                                  #Identifier of requested fields
                     “format”: {                                           #Format supported by the verifier for the set of elements
                                                                           requested in the input_descriptor that specifically contains it

                           “jwt_vc”: {
                                "alg": [“RS512“]
                           }
                     }
               }]
        }
}
```

11. Data model - QR presentation (request) from content provider

Body of the evidence

```
{
      “vp_token”: {                                                    #verifiable presentation
         "@context": "https://www.w3.org/ns/credentials/v2",
         "id":"data:application/vp+ld+json+jwt;${presentacionVerificableJWT}”,
         "type": "EnvelopedVerifiablePresentation"
      },
      “presentation_submission”: {
            "id": "a30e3b91-fb77-4d22-95fa-871689c322e2",
            "definition_id": “32f54163-7166-48f1-93d8-f f217bdb0653”,       #id of the presentation definition
            "descriptor_map": [ {
                        "id": “Age over 18”,                                    #id input descriptor
                        "format": "jwt_vc",
                        "path": "$.verifiableCredential[0]”
            }]
      },
      “nonce”: “07d54d63-7136-3ff1-11d8-f 9d17bdb0620”                        #Nonce field of the authorization request, used to manage
                                                                        the session and ensure that the presentation is not reused
}
```

Evidence signed by the holder of the credential

The evidence is secured by signing the body of the authorization response with the end user’s private key, thus ensuring that **even if the evidence is intercepted** and a nonce is requested from the content provider, **a valid authorization response cannot be sent** since the private key of the credential holder included in the evidence is not possessed.

Return to the evidence

Thank you