Issuer API Specification

Version 1.12

September 2022

A Digital India Initiative
National e-Governance Division.
Department of Electronics and Information Technology.

Revision History

Version	Date	Comments
1.0	15/01/2016	Release of version 1.0
1.1	11/04/2016	Added DocType element in Pull URI Request API
1.2	01/06/2016	Added Aadhaar related parameter details
1.3	20/07/2016	Added support to accept certificate metadata in Pull Document API.
1.4	29/11/2017	Added support to share Aadhaar photograph in Pull URI Request API and introduced a Pending response status in Pull Response.
1.5	19/07/2018	Updated Pull URI Request and Pull Doc Request to support machine readable certificate data.
1.6	07/08/2018	Updated the description of Pull URI Request API for Aadhaar based Pull functionality. Added API configuration details.
1.7	30/10/2018	Added Aadhaar, Name and DOB parameters in Pull Document API.
1.8	12/08/2019	Changed DataContent tag to include base64 encoded xml content of certificate data.
1.9	26/06/2020	Added DigiLockerId as a default parameter to Pull URI Request and Pull Document Request APIs.
1.10	04/08/2020	Added x-digilocker-hmac parameter in Pull URI and Pull Doc Requests for authentication.
1.11	26/03/2021	Added Name, DoB, Gender, Mobile and Photograph in Pull URI Response API so that the name matching can be performed on DigiLocker side.
1.12	15/09/2022	DataContent has become mandatory. Removed Aadhaar, Name and DOB parameters from Pull Document API.

Table of Contents

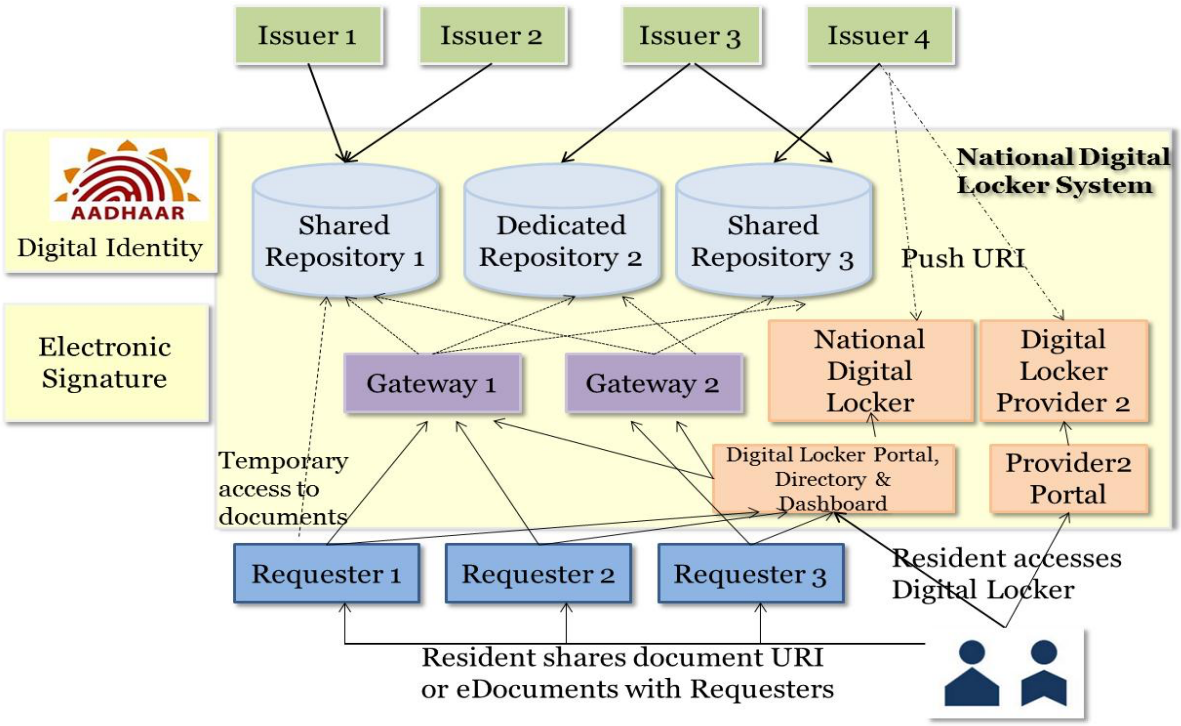
Digital Locker Issuer API Specification

1. Introduction

This document provides detailed specification of the Digital Locker Pull APIs. The Pull model of integration with Digital Locker enables a Digital Locker user to search a document/certificate from issuer repository and fetch (pull) it into Digital Locker. The issuer departments can use these APIs for the documents that are not Aadhaar seeded. For Aadhaar seeded documents, please refer to Dedicated Repository API Specification of Digital Locker. This document assumes that the reader is aware of Digital Locker application functionality and has read the Digital Locker Technical Specification (DLTS) available in Technical Specification section of Digital Locker Resource Center at https://digitallocker.gov.in/resource-center.php.

2. Digital Locker System Overview

The proposed architecture of the Digital Locker system is described in "Digital Locker Technical Specifications (DLTS)" document. Digital Locker system consists of e-Documents repositories and access gateways for providing an online mechanism for issuers to store and requesters to access a Digital Document in a uniform way in real-time.



3. Key Terminology

1. ***Electronic Document or E-Document*** – A digitally signed electronic document in XML format issued to one or more individuals (Aadhaar holders) in appropriate format compliant to DLTS specifications. Examples:
    - Degree certificate issued to a student by a university.
    - Caste certificate issued to an individual by a state government department.

- Marriage certificate issued to two individuals by a state government department.

2. ***Digital Repository*** **–** A software application complying with DLTS specifications, hosting a collection (database) of e-documents and exposing a standard API for secure real-time access.
    - While architecture does not restrict the number of repository providers, it is recommended that few highly available and resilient repositories be setup and encourage everyone to use that instead of having lots of repositories.

3. ***Digital Locker*** **–** A dedicated storage space assigned to each resident, to store authenticated documents. The digital locker would be accessible via web portal or mobile application.

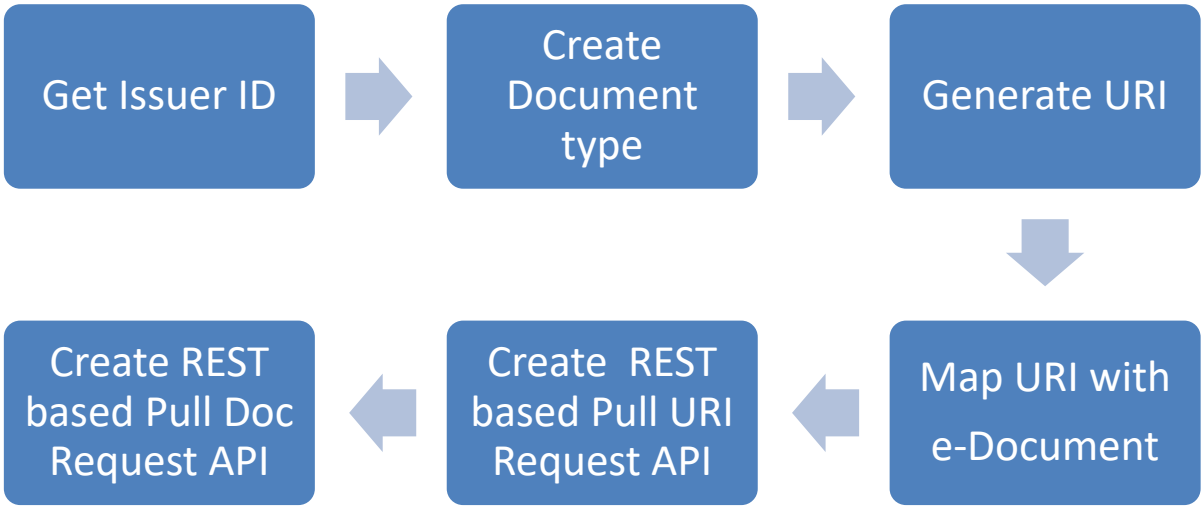
4. ***Issuer*** **–** An entity/organization/department issuing e-documents to individuals in DLTS compliant format and making them electronically available within a repository of their choice.

5. ***Requester*** **–** An entity/organization/department requesting secure access to a particular e-document stored within a repository. Examples:
    - A university wanting to access 10th standard certificate for admissions
    - A government department wanting to access BPL certificate
    - Passport department wanting to access marriage certificate

6. ***Access Gateway*** **–** A software application complying with DLTS specifications providing an online mechanism for requesters to access an e-document in a uniform way from various repositories in real-time.
    - Gateway services can be offered by repository providers themselves.
    - While architecture does not restrict the number of repository providers, it is suggested that few resilient and highly available central gateway systems be setup and requesters can signup with any one of the gateways for accessing documents in the Digital repositories.

7. ***Document URI*** **–** A unique document URI mandatory for every document. This unique URI can be resolved to a full URL to access the actual document in appropriate repository.
    - Document URI is a persistent, location independent, repository independent, issuer independent representation of the ID of the document.
    - The existence of such a URI does not imply availability of the identified resource, but such URIs are required to remain globally unique and persistent, even when the resource ceases to exist or becomes unavailable.
    - While document URI itself is not a secret, access to the actual document is secure and authenticated.

4. On-Boarding Flow



5. Document Codification Scheme

5.1 Unique Document URI

Every document that is issued and made accessible via DigiLocker must have a unique way to resolve to the correct repository without conflict. This is critical to eliminate the need for all documents reference to be in one system. Federated repositories storing documents issued by various departments/agencies must be "reachable" via the gateway in a unique fashion.

All documents issued in compliance to DLTS should have the following URI format:
`IssuerId-DocType-DocId` where

`IssuerId` is a unique issuer entity ID across the country
`DocType` is the document type optionally defined by the issuer
`DocId` is a unique document ID within the issuer system

5.2 Issuer ID (mandatory)

All departments/agencies within government issuing citizen documents, termed as "*Issuers*" must have a unique identification to ensure all documents issued by them are accessible via DLTS gateway.

It is recommended that list of unique issuer codes be derived via their domain URL whenever available and be published as part of e-governance standard codification scheme with ability to add new issuers on need basis. When URL is not available for a department, a unique (alpha) code may be assigned.

Examples of issuer Ids are "maharashtra.gov.in" (Maharashtra State Government), "kseeb.kar.nic.in" (Karnataka School Board", "cbse.nic.in" (CBSE School Board), "UDEL" (Delhi University), etc. These codes **MUST BE unique across India** and published as part of standard e-governance codification list.

5.3 Document Type (mandatory)

Issuers can freely define a list of document types for their internal classification. For example, CBSE may classify certificates into "MSTN" (10th mark sheet), "KVPY" (certificate issued to KVPY scholarship fellows), etc. There are no requirements for publishing these via any central registry.

Classifying documents into various types allows issuers to choose different repositories for different types. This is to future proof the design without making assumption that all certificates issued by the issuer are available in same repository. This also allows migration from one repository to another in a gradual way. Issuers are free to define their document types without worrying any collaboration across other issuers. Keeping the length minimal allows manual entry of document URI without making it too long. Hence it is recommended to keep length to be only up to 5.

It is recommended that issuers define document types either using pure alpha case-insensitive strings of length up to 5. These document types MUST BE unique WITHIN the issuer system. This classification within the issuer system also allows versioning of documents making future documents to be of different formats and in different repositories without having the need to have all documents in one repository. **If need arises in future to go beyond length 5, maximum length of doc type can easily use increased without breaking compatibility any existing systems and documents.**

5.4 Document ID (mandatory)

A document ID determined by the department/agency (issuer) should be assigned to every document. It MUST BE unique either within the document types of that issuer or it can be unique across all document types of that issuer.

Document ID is an alpha-numeric string with maximum length of 10. It is recommended that issuers define document IDs either using pure alpha case-insensitive string using a RANDOM number/string generator. Document IDs MUST BE unique WITHIN the issuer system within a document type. If need arises in future to go beyond length 10, maximum length of doc ID can easily use increased without breaking compatibility any existing systems and documents.
Using random string eliminates the possibility of "guessing" next sequence number and accessing a list of documents in a sequential way. This is critical to ensure security of documents and ensures document can be accessed ONLY IF the requester "knows" the actual document ID (instead of guessing sequential numbers).
It is highly recommended that issuer needing to issue a total of ***n*** documents within a document type use at least ***10n*** random space from which the strings/numbers are chosen to randomly allocate. Notice that since document types allow further classification, it is suggested to keep the length **minimal**. Since issuers can easily add a new document type without any collaboration and approvals across other issuers, if more numbers are required, a new document type may be introduced.

6. Document Issuance Flow

Document issuance flow is given below:

1. Create a digitally signed e-document complying to DLTS specification with a unique URI .
   a. Issuer entity uses the unique code for itself (obtain a new one if not already listed) that is available in common DLTS Issuer Codification e-governance standards. This is a country wide "Unique Issuer ID".
   b. Document type codification is done by the Digital Locker system administrator. Issuers may choose an available document type or if a new type of document is being issued then request Digital Locker team to create the required document type.
2. Issuer should create a document repository for storing documents and making it available online. This could be an existing database or document management system where the issued documents are stored.
3. Issue the printed document to the individual(s) for whom the document is issued to with a human readable document URI.
   a. Issuer should also offer an option to people to push the document URI to the digital lockers of the resident for whom the document was issued.

7. E-Document Specifications

7.1 Document URI

All documents issued in compliance to DLTS should have the following URI format:

`<IssuerId>[-DocType]-<DocId>`

Where,

`IssuerId` (mandatory) - is a unique issuer entity ID. This is a unique pure alpha case-insensitive string. To easily make it unique, department's domain URL can be used whenever available. The list of issuer Ids must be published and should have a mechanism to add new ones as required. **Unique list of Issuer IDs MUST BE unique and published via central e-governance codification scheme**.

`DocType` (mandatory) - is the document type optionally defined by the issuer. This is highly recommended for document classification and versioning purposes. Issuers may decide their own classification mechanism. This is a 5 char pure alpha string which can be expanded in future as needed.

`DocId` (mandatory) - is a unique document ID of length up to 10 within the issuer system. It is highly recommended that this is either purely numeric or alpha to avoid confusion with "0" with "o" etc. Also, it is highly recommended to use random strings to avoid guessing the sequence of document IDs.

7.2 Document Owner

DigiLocker ensures that the individual can access the document from issuer's repository only when the owner uniquely identifies a document that belong to him/her and the individual's profile data matches with the document data in the issuer's repository. This ensures that the documents are not misused.

7.3 Document Format

All e-documents must be represented in PDF or XML format complying to DLTS specifications. This ensures that a standardized XML structure is used to capture common attributes of all documents.

8. Issuer APIs

The issuer organization integrating with Digital Locker maintains the documents/certificates in its own repository (database or file system). The issuer application provides APIs to Digital Locker to access the documents in this repository. Each issuer organization will have to implement 2 APIs to integrate with the Digital Locker system. These 2 APIs are:

1. Pull URI Request API: This REST based pull API has to be implemented by the issuer organization to allow a locker owner to query the issuer repository by providing his/her Aadhaar number or any other identifier applicable to issuer organization (such as Roll number + Year + Class for CBSE mark sheet). This way the issuer may provide the URI of the document that is linked to the Aadhaar number or other identifiers provided by the resident.

2. Pull Doc Request API: This REST based pull API has to be implemented by the issuer organization to allow a resident to fetch a document from the issuer repository by providing the URI of the document.

These 2 APIs are defined in greater details in subsequent sections.

8.1 Pull URI Request API

The REST based Pull URI Request API has to be implemented by the issuers and will be consumed by Digital Locker application. This API will be invoked when a citizen searches the issuer repository for his/her certificate. If the certificate data is Aadhaar seeded, the issuer may choose to use Aadhaar number as the search parameter. Digital Locker provides Aadhaar number, name and date of birth as on Aadhaar to the issuer API as additional parameters. The option for these Aadhaar based parameters can be selected while configuring this API in Digital Locker Partner's Portal. If the certificate data is not Aadhaar seeded then the issuer may use any other unique parameter e.g. driving license number to search for a driving license. These custom parameters will be passed in the UDF elements as shown in the sample request below. The custom parameter(s) can be configured while configuring the API in the DigiLocker Partner's Portal. The Digital Locker system will query the issuer repository to fetch the URI for any document that match the search criteria. The citizen can save this URI in his/her Digital Locker. It is strongly recommended that the issuer API validate that the name, date of birth details sent by DigiLocker in Aadhaar parameters match with the corresponding details on the certificate before returning the certificate data. This will ensure that only authentic owners get access to a certificate.

8.1.1 Pull URI Request Format

HTTP Method: POST

HTTP Request Header Parameters:

- Content-Type: application/xml
- x-digilocker-hmac: This is used for authentication and to verify the integrity of the request. DigiLocker calculates the hash message authentication code (hmac) of the HTTP request body using SHA256 hashing algorithm and the API Key provided by the

issuer as the hashing key. The API Key is specified by the issuer while configuring the Pull URI API in DigiLocker Partner Portal. The resulting hmac is converted to Base64 format and sent in this parameter. It is strongly recommended that the issuer API calculates the hmac of the HTTP request body, convert it to Base64 and match it with this parameter to ensure authenticity of the request.

The following is the XML request template for the PULL URI Request API.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<PullURIRequest xmlns:ns2="http://tempuri.org/" ver="1.0" ts="YYYY-MM-
DDThh:mm:ss+/-nn:nn" txn="1234" orgId="" format="xml/pdf/both">
 <DocDetails>
    <DocType></DocType> //Document type
    <DigiLockerId></DigiLockerId > //Unique 36 character DigiLocker Id
    <UID></UID> //MD5 Hash of Aadhaar Number (Optional)
    <FullName> </FullName> //Name as on Aadhaar (Optional)
    <DOB></DOB> //Date of birth as on Aadhaar (Optional)
    <Photo></Photo> //Base 64 encoded JPEG photograph as on Aadhaar
(Optional)
    <UDF1></UDF1> //User defined field
    <UDF2></UDF2> //User defined field
    <UDF3></UDF3> //User defined field
        ...
    <UDFn></UDFn> //User defined field
</DocDetails>
</PullURIRequest>
```

8.1.2 Pull URI Request Elements

Various elements/attributes in the request are described below-

Sr. No.	XML Element	Mandatory (M)/ Optional (O)	Description
1.	ver	M	API version.
2.	ts	M	A timestamp value. This will be used to decode the keyHash element described below.
3.	txn	M	Transaction id.
4.	orgId	M	Org Id is the user id provided to the Digital Locker application by the issuer application for accessing the API.
5.	format	M	Indicates the desired format of the certificate data in the response. Possible values of this attribute are: **xml (default value)**: for certificate data in machine readable xml format

			both: for certificate data in both xml and pdf format. Please see the response section below for more details.
6.	DocType	M	Digital Locker will pass the document type being searched in this parameter.
7.	DigiLockerId	M	A unique 36 character DigiLocker Id of the user account.
8.	UID	O	MD5 Hash of Aadhaar Number of the DigiLocker user searching for the document/certificate. This is an optional parameter and will be sent only if the issuer opts for it while configuring the API on Digital Locker Issuer Portal.
9.	FullName	O	Name of the DigiLocker user searching for the document/certificate as on Aadhaar. This is an optional parameter and will be sent only if the issuer opts for it while configuring the API on Digital Locker Issuer Portal.
10.	DOB	O	Date of birth of the DigiLocker user searching for the document/certificate as on Aadhaar in DD-MM-YYYY format. This is an optional parameter and will be sent only if the issuer opts for it while configuring the API on Digital Locker Issuer Portal.
11.	Photo	O	The base 64 encoded contents of JPEG photograph as on Aadhaar. This is an optional parameter and will be sent only if the issuer opts for it while configuring the API on Digital Locker Issuer Portal.
12.	UDF1…n	M	User defined search parameters to search a unique document/certificate. The <UDF> may be <RollNo> for CBSE, <RegistrationNo> for Transportation Dept. and <PAN> for Income Tax Dept. The search parameters for the API will be configured in the issuer portal of Digital Locker while configuring this API.

8.1.3 Pull URI Response Format

The response to the Pull URI request will include the URI of the document linked to the given search criteria in the request as well as the base 64 encoded data of the document. The issuer will provide the response back to the Digital Locker system synchronously.

The following is the XML response template for the Pull URI Response API.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

  <PullURIResponse xmlns:ns2="http://tempuri.org/">
    <ResponseStatus Status="1" ts="2016-01-11T14:44:48+05:30"
txn="1452503688">1</ResponseStatus>//1-Success //0-Failure //9-Pending
    <DocDetails>
        <IssuedTo>
          <Persons>
            <Person name="Sunil Kumar" dob="31-12-1990"
gender="Male/Female/Transgender" phone="9876543210">
              <Photo format=" PNG/JPG/JPEG">Base 64 encoded image
content</Photo>
            </Person>
            <Person name="Sunita Devi" dob="25-03-1993"
gender="Male/Female/Transgender" phone="9873451238"/>
          </Persons>
        </IssuedTo>
        <URI>in.gov.dept.state-INCER-1234567</URI>
        <DocContent>
               //Base64 encoded string of PDF file
        </DocContent>
        <DataContent>
               //Base64 encoded certificate metadata in XML format
        </DataContent>
    </DocDetails>
  </PullURIResponse>
```

8.1.4 Pull URI Response Elements

Various elements/attributes in the response are described below-

Sr. No.	XML Element	Mandatory (M)/ Optional (O)	Description
1.	ts	M	A timestamp value as sent in the request.
2.	txn	M	Transaction id value as sent in the request.
3.	Status	M	1 for success, 0 for error and 9 for pending.

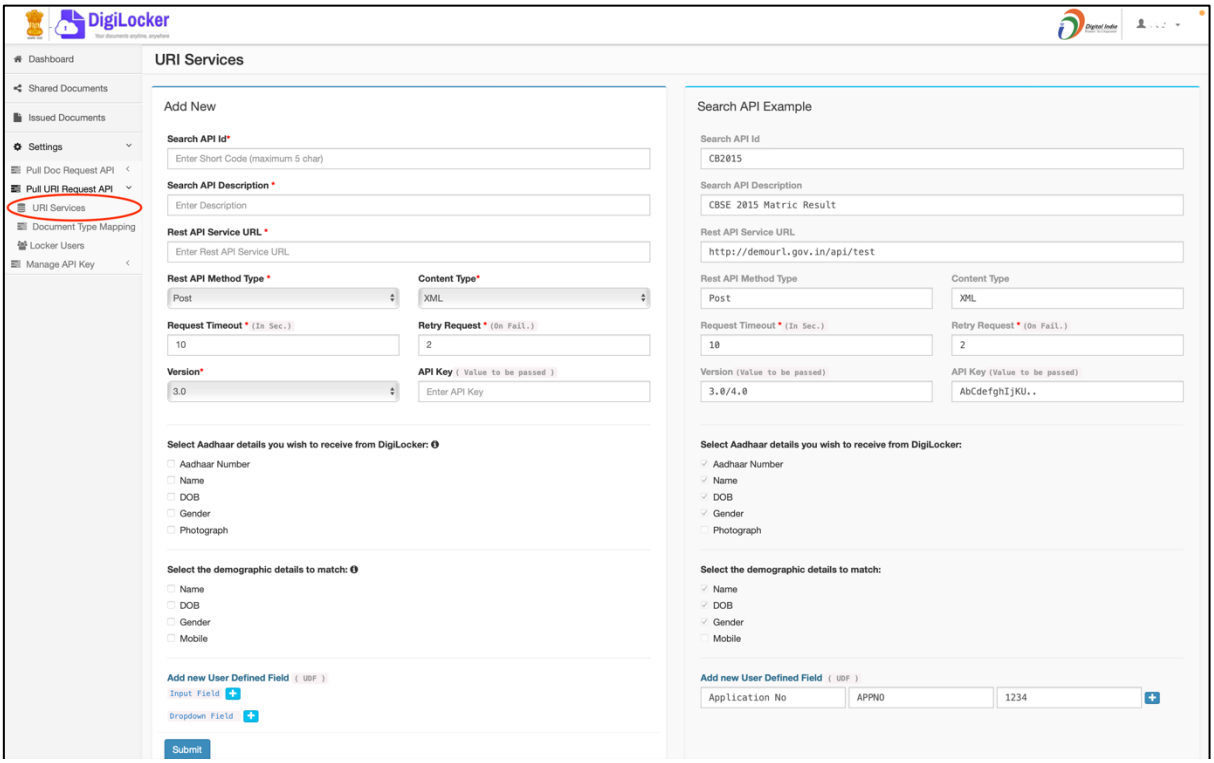
			The issuer department may do a manual verification of the Pull Request and take a decision to provide a document at a later time. In this case the response status should contain value 9. DigiLocker will show an appropriate message to the user in this case. Upon successful verification, the issuer department can use PUSH URI API as mentioned in Digital Locker Dedicated Repository API Specification.
4.	DocDetails	M	Issuer can add meta content specific to document here.
5.	IssuedTo	M	Contains the details about the individual/s to whom the certificate is issued. It contains one or more Person/s elements.
6.	Persons	M	Contains the details about the individual/s to whom the certificate is issued. It contains one or more Person/s elements. If you select, *Match Name, DoB, Gender or Mobile* option while configuring Pull URI API in DigiLocker Partner's Portal, DigiLocker matches the corresponding details of each individual in this list with the name in DigiLocker KYC profile. If the information of any one of the individual matches, DigiLocker will provide the certificate to the individual.
7.	Person	M	Contains the details about an individual to whom the certificate is issued.
8.	name	M	The name of individual as on the document as per the issuer's record. If you select, *Match Name* option while configuring Pull URI API in DigiLocker Partner's Portal, DigiLocker uses this name to match with the name in DigiLocker KYC profile.
9.	dob	M	The date of birth of individual as on the document as per the issuer's record in DD-MM-YYYY

			format. If you select, *Match DoB* option while configuring Pull URI API in DigiLocker Partner's Portal, DigiLocker uses this field to match with the DoB in DigiLocker KYC profile.
10.	gender	M	The gender of individual as on the document as per the issuer's record. If you select, *Match Gender* option while configuring Pull URI API in DigiLocker Partner's Portal, DigiLocker uses this field to match with the Gender in DigiLocker KYC profile. The possible values for this field are Male, Female or Other
11.	phone	M	The mobile number of individual as on the document as per the issuer's record. If you select, *Match Mobile* option while configuring Pull URI API in DigiLocker Partner's Portal, DigiLocker uses this field to match with the mobile number in DigiLocker profile.
12.	Photo	O	Contains the base64 encoded content of photograph in PNG, JPG or JPEG format of the individual to whom the certificate is issued.
13.	format	M	Format of the photograph. It will contain one of the following values: PNG JPG JPEG
14.	URI	M	URI corresponding to the search criteria that identifies the document uniquely.
15.	DocContent	O	Enclose the Base64 byte encoded contents of PDF file in this element. The DocContent element should be sent only if the *format* attribute in the original request is sent as "both".
16.	DataContent	M	Enclose the base64 byte encoded certificate metadata in XML format.

8.1.5 Configuration of Pull URI API in DigiLocker Partner Portal

Once you have developed and deployed the API on your server, the next step is to provide the details of this API to DigiLocker so that DigiLocker can call the API. For this, login in to

DigiLocker Partner Portal using your issuer credentials. On the left menu in your account click on Settings->Pull URI Request API-> URI Services. You will see an 'Add' button on the page displayed on the right side. Click on the 'Add' button to add a new API. Configure the details of your API in the page displayed and click 'Submit'. You can also add the details of the user defined parameters (UDFs) if you are using custom parameters to search a document.



With this step, DigiLocker now knows about the end point of your API along with the parameter it takes. DigiLocker uses these configuration details to display the search document screen to a user. Please see the image above for more details.

8.2 Pull Doc Request API

The REST based Pull Doc Request API has to be implemented by the issuers and will be consumed by Digital Locker system. This API will be invoked when the resident clicks on the URI displayed in the Govt. Issued documents section of the Digital locker portal. The issuer system will respond to this API by sending the certificate data. The certificate data should be sent in one of the two formats depending on the request send by Digital Locker:

a. PDF document format
b. XML format for machine readable metadata

8.2.1 Pull Doc Request Format

HTTP Method: POST

HTTP Request Header Parameters:

- Content-Type: application/xml
- x-digilocker-hmac: This is used for authentication and to verify the integrity of the request. DigiLocker calculates the hash message authentication code (hmac) of the HTTP request body using SHA256 hashing algorithm and the API Key provided by the

issuer as the hashing key. The API Key is specified by the issuer while configuring the Pull Doc API in DigiLocker Partner Portal. The resulting hmac is converted to Base64 format and sent in this parameter. It is strongly recommended that the issuer API calculates the hmac of the HTTP request body, convert it to Base64 and match it with this parameter to ensure authenticity of the request.

The following is the XML request template for the PULL Doc Request API.

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<PullDocRequest xmlns:ns2="http://tempuri.org/" ver="1.0" ts="YYYY-MM-
DDThh:mm:ss+/-nn:nn" txn="" orgId="" format="xml/pdf/both">
  <DocDetails>
    <URI>in.gov.kerala.edistrict-INCER-123456</URI>
    <DigiLockerId>123e4567-e89b-12d3-a456-426655440000</DigiLockerId>
//Unique 36 character DigiLocker Id
  </DocDetails>
</PullDocRequest>
```

8.2.2 Pull Doc Request Elements

Various elements/attributes in the request are described below-

Sr. No.	XML Element	Mandatory (M)/ Optional (O)	Description
1.	ver	M	API version.
2.	ts	M	A timestamp value. This will be used to decode the keyHash element described below.
3.	txn	M	Transaction id.
4.	orgId	M	Org Id is the user id provided to the Digital Locker application by the issuer application for accessing the API.
5.	format	M	Indicates the desired format of the certificate data in the response. Possible values of this attribute are: **xml (default value)**: for certificate data in machine readable xml format **both**: for certificate data in both xml and pdf format. Please see the response section below for more details.
6.	URI	M	URI identifies the document uniquely.
7.	DigiLockerId	M	A unique 36 character DigiLocker Id of the user account.

8.2.3 Pull Doc Response Format

The response to the PULL Doc request will include the Doc content of any documents linked to the given URI in the request. The issuer will provide the response back to the Digital Locker system synchronously. The PDF data should be sent in DocContent element and XML metadata should be sent in MetadataContent element. The response should contain the certificate data in only one of these formats based on the *metadata* attribute in the request.

The following is the XML response template for the PULL Doc Response API.

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<PullDocResponse xmlns:ns2="http://tempuri.org/">
        <ResponseStatus Status="1" ts=" YYYY-MM-DDThh:mm:ss+/-nn:nn"
txn=""> //1-Success //0-Failure
        </ResponseStatus>
        <DocDetails>
                //Send one of DocContent or MetadataContent element
based on the metadata attribute in the request.
                <DocContent>
                        //Bytes encoded with Base64 in string format
                </DocContent>
                <DataContent>
                        //Base64 encoded certificate metadata in XML
format
                </DataContent>
        </DocDetails>
</PullDocResponse>
```

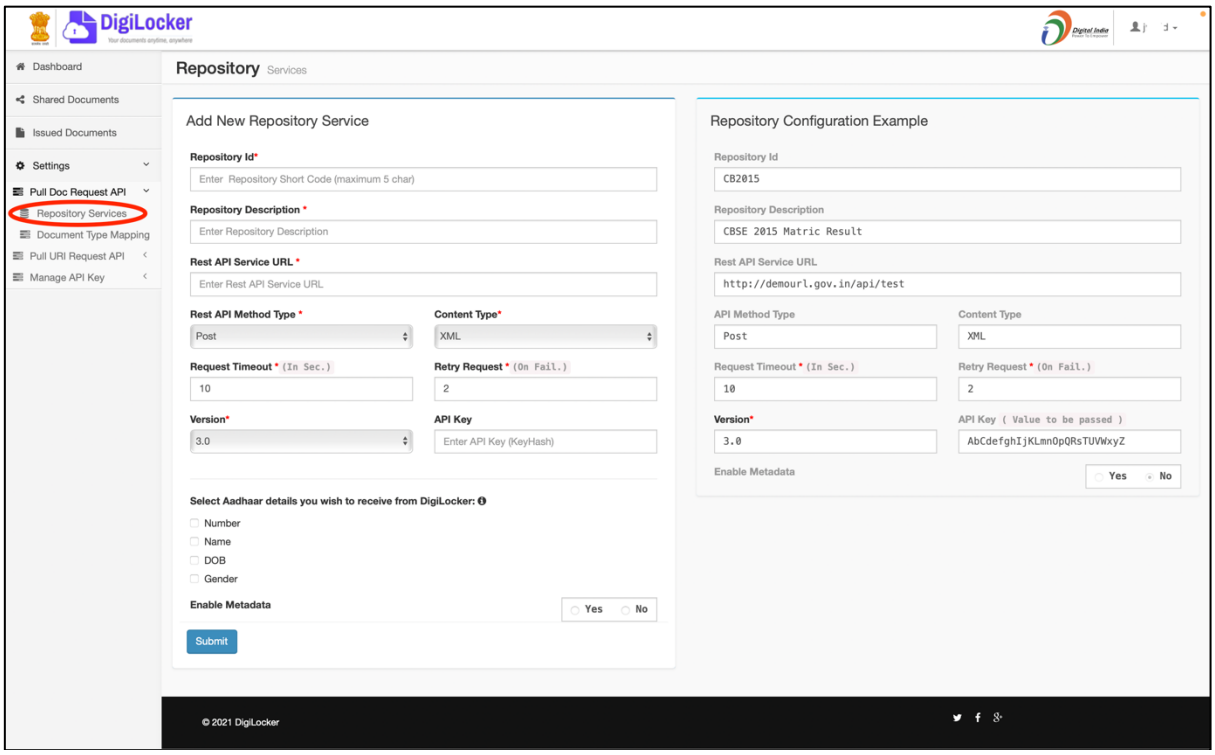
8.2.4 Pull Doc Response Elements

Various elements/attributes in the response are described below-

Sr. No.	XML Element	Mandatory (M)/ Optional (O)	Description
1.	Ts	M	A timestamp value as sent in the request.
2.	txn	M	Transaction id value as sent in the request.
3.	Status	M	1 for success, 0 for error.
4.	DocDetails	M	Issuer can add meta content specific to document here.
5.	DocContent	O	Enclose the Base64 byte encoded contents of PDF file in this element. The DocContent element should be sent only if the *format* attribute in the original request is sent as "both".

6.	DataContent	M	Enclose the Base64 byte encoded certificate metadata in XML format.

8.2.5 Configuration of Pull Doc API in DigiLocker Partner Portal



Once you have developed and deployed the API on your server, the next step is to provide the details of this API to DigiLocker so that DigiLocker can call the API. For this, login in to DigiLocker Partner Portal using your issuer credentials. On the left menu in your account click on Settings->Pull Doc Request API API->Repository Services. You will see an 'Add' button on the page displayed on the right side. Click on the 'Add' button to add a new API. Configure the details of your API in the page displayed and click 'Submit'. With this step, DigiLocker now knows about the end point of your API along with the parameter it takes. DigiLocker uses these configuration details to fetch a document using URI when a user clicks on an issued document in his/her DigiLocker account. Please see the image above for more details.