

Piattaforma Digitale Nazionale Dati

Documentazione tecnica e-service di Regione Lombardia

ET_TO_BO - RL GATEWAY

Versione 1.0.0

Indice

1. Introduzione.....	3
2. Autenticazione	3
3. Descrizione degli ambienti.....	3
4. Metodi di chiamata.....	3
4.1 Metodo Retry.....	3
4.1.1 Parametri di Richiesta	4
4.1.2 Esempio di risposta	4
4.2 Metodo Send instance	5
4.2.1 Parametri di Richiesta	5
4.2.2 Esempio di risposta	7
4.3 Metodo Resource.....	7
4.3.1 Parametri di Richiesta	7
4.3.2 Esempio di risposta	7
4.4 Metodo Notify.....	8
4.4.1 Parametri di Richiesta	8
4.4.2 Esempio di risposta	8
5. Uso dei codici di stato HTTP.....	10

1. Introduzione

Il servizio permette di gestire le istanze dei procedimenti SUAP di competenza dei servizi del SIRE di Regione Lombardia (ARPA, etc.), avviate su richiesta dei Soggetti presentatori, fungendo da gateway tra la componente informatica Enti terzi e il BackOffice SUAP.

Per qualsiasi esigenza relativa all'e-service è possibile richiedere assistenza alla seguente mail:
assistenza_api@regione.lombardia.it .

2. Autenticazione

L'autenticazione sarà gestita attraverso la piattaforma PDND che genera un voucher di 30 minuti e di tipo Bearer JWT. Tale e-service non prevede ulteriori pattern di interoperabilità MODI.

3. Descrizione degli ambienti

Gli URL Base rispettivamente dell'ambiente di collaudo e produzione in PDND sono i seguenti:

- URL Base di Collaudo: <https://api.integrazione.lispa.it/c/pdndbasic/erogazione/hub-core/api/out/v1.0.0>
- URL Base di Produzione: <https://api.servizirl.it/c/pdndbasic/erogazione/hub-core/api/out/v1.0.0>

4. Metodi di chiamata

Di seguito sono descritti i metodi messi a disposizione del servizio con i relativi parametri da valorizzare.

4.1 Metodo Retry

La funzione permette di richiedere la ritrasmissione dati relativi ad una specifica operation a causa di incoerenze riscontrate sui dati ricevuti.

- ENDPOINT: Url Base + “/retry”
- METODO: POST
- HEADER:
 - Content-Type: application/Json
 - Authorization: Bearer Token_Di_Acesso
 - AgID-JWT-Signature: Token JWT firmato per la verifica integrità del messaggio

4.1.1 Parametri di Richiesta

Si riporta di seguito un esempio di body della request

```
{
  "cui" : {
    "context" : "SUAP",
    "progressivo" : "100",
    "data" : "2025-06-26",
    "uuid" : "80bc7d1a-10e5-11ee-be56-0242ac120003"
  },
  "operation" : "request_integration",
  "error" : {
    "code" : "",
    "message" : ""
  }
}
```

Dove i campi sono definiti nel seguente modo:

```
{
  "cui": {
    "context": "string",
    "data": "yyyy-mm-dd",
    "progressivo": "string",
    "uuid": "string"
  },
  "operation": "send_conclusions",
  "error": {
    "code": "string",
    "message": "string"
  }
}
```

operation può essere una delle seguenti:

- send_conclusions
- request_cdss
- request_integration

error può assumere uno dei seguenti valori (coppie code–message):

code	message
ERR02	Documento non trovato
ERR03	La pratica non esiste
ERR04	Errore interno
ERR06	Parametri errati o mancanti
ERR10	Errore generico

4.1.2 Esempio di risposta

Nei casi di esito positivo la chiamata restituisce un codice HTTP 200 (OK) con body vuoto.

In caso di errore gestito dal backend, viene invece restituita una response in formato JSON contenente i dettagli dell'errore

```
{
  "code": "string",
  "message": "string"
}
```

Dove:

Nome del campo	Tipo dato	Note
code	string	Contiene il codice HTTP dell'errore.
message	string	Contiene la descrizione specifica dell'errore.

4.2 Metodo Send instance

La funzione permette la ricezione di una istanza di un procedimento SUAP.

- ENDPOINT: Url Base + “/send_instance”
- METODO: POST
- HEADER:
 - Content-Type: application/Json
 - Authorization: Bearer Token_Di_Acesso
 - AgID-JWT-Signature: Token JWT firmato per la verifica integrità del messaggio

4.2.1 Parametri di Richiesta

Si riporta di seguito un esempio di body della request

```
{
  "cui": {
    "context": "SUAP",
    "data": "2025-08-08",
    "progressivo": "2139",
    "uuid": "2e92ad65-7e49-42ea-9306-c1fd03c2e770"
  },
  "instance_descriptor_version": "0",
  "instance_index": [
    {
      "code": "USEC-0001427",
      "ref": "02313821007-08082025-1040.002.MDA.XML",
      "resource_id": "02313821007-08082025-1040.002.MDA.XML",
      "hash": "7c8cf57e06f3e8ec0d1dc04eab3ab93ceb4fd18c56cb919cb4195de9b36ce42",
      "alg_hash": "S256"
    }
  ],
  "general_index": [
    {
      "name": "SUAP_PDF",
      "mime_type": "application/pdf",
      "resource_id": "02313821007-08082025-1040.SUAP.PDF.P7M",
      "hash": "e3407367e28532b62c56dc62ae470a989a1f49d29d168b597fd5ed3d07a8c7e7",
      "alg_hash": "S256"
    }
  ]
}
```

Dove i campi sono definiti nel seguente modo:

```
{
  "cui": {
    "context": "string",
    "data": "yyyy-mm-dd",
    "progressivo": "string",
    "uuid": "string"
  },
  "instance_descriptor_version": "string",
  "instance_index": [
    {
      "code": "string",
      "ref": "string",
      "resource_id": "string",
      "hash": "string",
      "alg_hash": "string"
    }
  ]
}
```

```
        "hash": "string",
        "alg_hash": "S256"
    },
    "general_index": [
        {
            "name": "RICEVUTA_XML",
            "mime_type": "application/xml",
            "resource_id": "string",
            "hash": "string",
            "alg_hash": "S256"
        },
        {
            "name": "RICEVUTA_PDF",
            "mime_type": "application/pdf",
            "resource_id": "string",
            "hash": "string",
            "alg_hash": "S256"
        },
        {
            "name": "SUAP_XML",
            "mime_type": "application/xml",
            "resource_id": "string",
            "hash": "string",
            "alg_hash": "S256"
        },
        {
            "name": "SUAP_PDF",
            "mime_type": "application/pdf",
            "resource_id": "string",
            "hash": "string",
            "alg_hash": "S256"
        },
        {
            "name": "SUAP_REA_START_XML",
            "mime_type": "application/xml",
            "resource_id": "string",
            "hash": "string",
            "alg_hash": "S256"
        },
        {
            "name": "SUAP_REA_UPDATE_XML",
            "mime_type": "application/xml",
            "resource_id": "string",
            "hash": "string",
            "alg_hash": "S256"
        },
        {
            "name": "SUAP_REA_END_XML",
            "mime_type": "application/xml",
            "resource_id": "string",
            "hash": "string",
            "alg_hash": "S256"
        },
        {
            "name": "COMUNICA_CUI_XML",
            "mime_type": "application/xml",
            "resource_id": "string",
            "hash": "string",
            "alg_hash": "S256"
        },
        {
            "name": "COMUNICA_CUI_PDF",
            "mime_type": "application/pdf",
            "resource_id": "string",
            "hash": "string",
            "alg_hash": "S256"
        }
    ]
}
```

4.2.2 Esempio di risposta

Nei casi di esito positivo la chiamata restituisce un codice HTTP 200 (OK) con body vuoto.

In caso di errore gestito dal backend, viene invece restituita una response in formato JSON contenente i dettagli dell'errore

```
{
  "code": "string",
  "message": "string"
}
```

Dove:

Nome del campo	Tipo dato	Note
code	string	Contiene il codice HTTP dell'errore.
message	string	Contiene la descrizione specifica dell'errore.

4.3 Metodo Resource

La funzione permette il recupero di una risorsa documento per un dato CUI. Le Range Request (RFC9110) sono supportate soltanto in modalità single part.

- ENDPOINT: Url Base + “/instance/{cui_uuid}/document/{resource_id}”
- METODO: GET
- HEADER:
 - Content-Type: application/Json
 - Authorization: Bearer Token_Di_Acesso
 - AgID-JWT-Signature: Token JWT firmato per la verifica integrità del messaggio
 - If-Match: Contiene l'hash del file ricevuto in precedenza dal Back Office (BO SUAP) nella risposta alle seguenti chiamate di servizio:
 - /coll-ssu_bo_to_et/request_integration
 - /coll-ssu_bo_to_et/send_conclusions

4.3.1 Parametri di Richiesta

Nome	Tipo	Obblig.	Case sensitive	Descrizione	Valori amiss.	Esempio
cui_uuid	string	SI	SI	UUID del CUI		80bc7d1a-10e5-11ee-be56-0242ac120003
resource_id	string	SI	SI	Id risorsa documento		avvio vicinato.xml

4.3.2 Esempio di risposta

In caso di esito positivo, il servizio restituisce:

- Codice HTTP: 200 OK

- Body: contiene il file binario richiesto, codificato in Base64 all'interno del campo di risposta previsto.

In caso di errore o condizioni particolari, il servizio può restituire uno dei seguenti codici di stato HTTP, in base al tipo di anomalia riscontrata:

Codice	Descrizione
206	Contenuto parziale (partial content) – restituito quando la risposta contiene solo una parte del file richiesto.
400	Richiesta non valida (bad request) – i parametri della richiesta non sono corretti o incompleti.
401	Non autorizzato (unauthorized) – autenticazione mancante o non valida.
404	Risorsa non trovata (not found) – il file o l'identificativo richiesto non è disponibile.
412	Precondition failed – l'header If-Match non corrisponde o la precondizione indicata non è soddisfatta.
416	Range not satisfiable – l'intervallo di byte richiesto non è valido o fuori dal range disponibile.
428	Precondition required – l'header If-Match o altra precondizione obbligatoria è assente.
500	Errore interno del server.
503	Servizio non disponibile – il backend non è momentaneamente raggiungibile.

Nei casi di esito positivo la chiamata restituisce un codice HTTP 200 (OK) con il file binario codificato in Base64.

4.4 Metodo Notify

La funzione permette di notificare eventi di processo.

- ENDPOINT: Url Base + “/notify”
- METODO: POST
- HEADER:
 - Content-Type: application/Json
 - Authorization: Bearer Token_Di_Accesso
 - AgID-JWT-Signature: Token JWT firmato per la verifica integrità del messaggio

4.4.1 Parametri di Richiesta

Si riporta di seguito un esempio di body della request

```
{
  "cui": {
    "context": "SUAP",
    "data": "2025-01-03",
    "progressivo": "1",
    "uuid": "uuid_test_2025_01_23_1"
  },
  "instance_descriptor_version": "1.0",
  "event": "end_by_positive_outcome"
}
```

4.4.2 Esempio di risposta

Nei casi di esito positivo la chiamata restituisce un codice HTTP 200 (OK) con body vuoto.

In caso di errore gestito dal backend, viene invece restituita una response in formato JSON contenente i dettagli dell'errore.

```
{  
    "code": "string",  
    "message": "string"  
}
```

Dove:

Nome del campo	Tipo dato	Note
code	string	Contiene il codice HTTP dell'errore.
message	string	Contiene la descrizione specifica dell'errore.

5. Uso dei codici di stato HTTP

È importante utilizzare i codici di stato http, in quanto questi possono essere letti anche da utenti non umani.

Code (codice)	Code Summary	Descrizione
Succesful code		
200	OK	La richiesta è andata a buon fine.
201	Created	Indica che la richiesta HTTP ha portato alla creazione di una risorsa.
206	Partial Content	Indica che il server ha elaborato con successo una richiesta parziale del contenuto (specificata tramite l'header Range nella richiesta).
Error code		
400	Bad request	Il server non può elaborare la richiesta a causa di un errore del client (e.g. sintassi della richiesta non corretta: formato JSON non valido).
401	Unauthorized	Possibili due significati: a) L'autenticazione è stata richiesta e non è andata a buon fine; b) L'autenticazione non è ancora stata fornita.
403	Forbidden	Nessuna risorsa corrispondente trovata nell'API per la richiesta specificata.
412	Precondition Failed	La condizione specificata negli header della richiesta non è stata soddisfatta dal server.
415	Unsupported Media Type	L'entità richiesta ha un tipo di supporto che il server o la risorsa non supporta. Di seguito sono riportati i tipi di contenuto supportati: application/json, application/x-www-form-urlencoded e multipart/form-data. Se si invia un tipo di contenuto diverso dal tipo di contenuto supportato, viene visualizzato questo codice di stato.
416	Range Not Satisfiable	La porzione di contenuto richiesta tramite l'header Range non può essere soddisfatta dal server.
428	Precondition Required	Il server richiede che la richiesta includa una condizione preliminare (precondition) non presente.
429	Too many requests	Soglie di limitazione superate.

		L'intestazione http Retry-After specifica il numero di secondi di attesa prima di riprendere le chiamate API.
500	Internal server error	Si è verificato un errore del server.
501	Not implemented	Quando il server non riconosce il metodo di richiesta o non è in grado di soddisfare la richiesta.
503	Server unavailable	Risorse del server superate, servizio temporaneamente non disponibile.