

Signature Read

API di lettura per *Ts Digital Signature* per le operazioni riguardanti i documenti. [Swagger](#)

Header

Gli header richiesti dalla chiamata sono gli [header standard di TSDigital](#).

Il `Content-Type` deve essere `application/json`

Descrizione API

DOCUMENTS

Search documents

[POST] [/v1/documents/search](#)

Permette di effettuare una ricerca filtrata per i documenti di una determinata azienda

Query params

- `page` numero di pagina
- `size` numero di elementi per pagina
- `sort` campo per cui ordinare i documenti

Request body

- `page` numero di pagina **//TODO FIX**
- `size` numero di elementi per pagina **//TODO FIX**
- `sort` campo per cui ordinare i documenti **//TODO FIX**
- `managerId` identificativo dello studio a cui è connessa l'azienda, permette di visualizzare i documenti che lo studio ha creato per conto di una determinata azienda
- `documentStatusId` stato del documento
- `documentTypeId` tipo documento
- `documentOwnerId`

- ownerCriteria.ownerType
- ownerCriteria.searchTextCriteria
- batchId
- multiDocumentSessionId id del documento multi sessione creato
- documentIntermediaryId identificativo azienda, **campo obbligatorio**
- cctStatusCode stato della conservazione del documento
- lastTimestampFrom timestamp da cui inizierà la ricerca dei documenti che sono stati aggiornati fino al momento della chiamata
- minCreationDate data di creazione del documento da cui iniziare la ricerca
- maxCreationDate data di creazione del documento da per finire la ricerca
- expiryDate data di scadenza
- signerTextFieldsSearchCriteria permette di effettuare una ricerca per i firmatario inseriti nei documenti, il campi in cui viene fatta la ricerca sono //chiedere a Nico

```
{ //TODO FIX
  "page": {
    "page": 0,
    "size": 0,
    "sort": [
      "string"
    ]
  },
  "request": {
    "managerId": "string",
    "documentStatusId": "string",
    "documentTypeId": "string",
    "documentOwnerId": "string",
    "ownerCriteria": {
      "ownerType": "string",
      "searchTextCriteria": "string"
    },
    "batchId": "string",
    "multiDocumentSessionId": "string",
    "documentIntermediaryId": "string",
    "cctStatusCode": "string",
    "lastTimestampFrom": 0,
    "minCreationDate": "2021-02-24T23:34:28.076Z",
    "maxCreationDate": "2021-02-24T23:34:28.076Z",
    "expiryDate": "2021-02-24T23:34:28.076Z",
    "signerTextFieldsSearchCriteria": "string"
  }
}
```

```
}
```

Retrieve detail of single document

[GET] [/v1/documents/{hubId}](#)

Permette di recuperare tutte le info di un determinato documento

Query params

- `hubId` identificativo del documento

Get document signature links

[GET] [/v1/documents/{hubId}/signatureLink](#)

Restituisce il link riguardanti una determinata sessione in stato `ALLA_FIRMA` divisi per firmatari

Query params

- `hubId` identificativo del documento

Response

```
[
  {
    "fiscalCode": "CSAMRC80A01I829M",
    "email": "rmainini@mondora.com",
    "url": "https://b2bstaticdev.blob.core.windows.net/static-apps/b2b-firma/actions/redirect.html?sign_token=Y2NjNDYxZTgtOWEyNy000TFmLTkwYTQtZjd1Yzg1ODUzNmZl"
  },
  {
    "fiscalCode": "DBSNDR93T31B519E",
    "email": "rmainini@mondora.com",
    "url": "https://b2bstaticdev.blob.core.windows.net/static-apps/b2b-firma/actions/redirect.html?sign_token=MzFhZGE3NmItOTM2OS00YTFmLWE4MmEtNzg5ZTM5NmExODhl"
  }
]
```

Get multiSessionDocument signature links

[GET] </v1/documents/multiDocumentSession/{multiDocumentSessionId}/signatureLink>

Restituisce il link riguardanti una determinata sessione in stato `ALLA_FIRMA` divisi per firmatari per i documenti multi sessione

Query params

- `multiDocumentSessionId` identificativo del documento multi sessione

Response

```
[
  {
    "fiscalCode": "CSAMRC80A01I829M",
    "email": "rmainini@mondora.com",
    "url": "https://b2bstaticdev.blob.core.windows.net/static-apps/b2b-firma/actions/redirect.html?sign_token=Y2NjNDYxZTgtOWEyNy00OTdmLTkwYTQtZjd1Yzg1ODUzNmZl"
  },
  {
    "fiscalCode": "DBSNDR93T31B519E",
    "email": "rmainini@mondora.com",
    "url": "https://b2bstaticdev.blob.core.windows.net/static-apps/b2b-firma/actions/redirect.html?sign_token=MzFhZGE3NmItOTM2OS00YTZmLWE4MmEtNzg5ZTM5NmExODhl"
  }
]
```

Download attachment as base64

[GET] </v1/documents/{hubId}/attachment/{attachmentId}>

Permette il download in base64 di eventuali allegati del documento

Path params

- `hubId` identificativo del documento
- `attachmentId` identificativo dell'allegato, l'id dell'allegato si trova nel json del documento alla chiave **attachments[].id**

Download attachment

```
[GET] /v1/documents/{hubId}/attachment/{attachmentId}/file
```

Permette il download del file di eventuali allegati del documento

Path params

- `hubId` identificativo del documento
- `attachmentId` identificativo dell'allegato, l'id dell'allegato si trova nel json del documento alla chiave **attachments[].id**

Download base64 document

```
[GET] /v1/documents/{hubId}/download
```

Permette il download in base64 del documento

Path params

- `hubId` identificativo del documento

Query params

- `signed` se messo a true verrà scaricato il documento firmato sennò il documento originale

Download document

```
[GET] /v1/documents/{hubId}/download/file
```

Permette il download del file del documento

Path params

- `hubId` identificativo del documento

Query params

- `signed` se messo a true verrà scaricato il documento firmato sennò il documento originale

DOCUMENTS TYPES

Retrieve document type detail

[GET] [/v1/documentTypes/{id}](#)

Restituisce il dettaglio del singolo tipo di documento

Path params

- `id` identificativo del tipo di documento

Retrieve list of document types

[GET] [/v1/documentTypes](#)

Restituisce la lista dei tipi documento

DOCUMENTS STATUSES

Retrieve document status detail

[GET] [/v1/documentStatuses/{id}](#)

Restituisce il dettaglio del singolo stato documento

Path params

- `id` identificativo dello stato di un documento

Retrieve list of document types

[GET] [/v1/documentStatuses](#)

Restituisce la lista degli stati documento

ARCHIVED DOCUMENTS

Search archived documents

[POST] [/v1/archivedDocuments/search](#)

Permette di effettuare una ricerca filtrata per i documenti che sono stati cancellati di una determinata azienda

Query params

- `page` numero di pagina
- `size` numero di elementi per pagina
- `sort` campo per cui ordinare i documenti

Request body

- `page` numero di pagina **//TODO FIX**
- `size` numero di elementi per pagina **//TODO FIX**
- `sort` campo per cui ordinare i documenti **//TODO FIX**
- `managerId` identificativo dello studio a cui è connessa l'azienda, permette di visualizzare i documenti che lo studio ha creato per conto di una determinata azienda
- `documentStatusId` stato del documento
- `documentTypeId` tipo documento
- `documentOwnerId`
- `ownerCriteria.ownerType`
- `ownerCriteria.searchTextCriteria`
- `batchId`
- `multiDocumentSessionId` id del documento multi sessione creato
- `documentIntermediaryId` identificativo azienda, **campo obbligatorio**
- `cctStatusCode` stato della conservazione del documento
- `lastTimestampFrom` timestamp da cui inizierà la ricerca dei documenti che sono stati aggiornati fino al momento della chiamata
- `minCreationDate` data di creazione del documento da cui iniziare la ricerca
- `maxCreationDate` data di creazione del documento da per finire la ricerca
- `expiryDate` data di scadenza
- `signerTextFieldsSearchCriteria` permette di effettuare una ricerca per i firmatario inseriti nei documenti, il campi in cui viene fatta la ricerca sono **//chiedere a Nico**

```
{ //TODO FIX
  "page": {
    "page": 0,
    "size": 0,
    "sort": [
      "string"
    ]
  }
}
```

```
{,
  "request": {
    "managerId": "string",
    "documentStatusId": "string",
    "documentTypeId": "string",
    "documentOwnerId": "string",
    "ownerCriteria": {
      "ownerType": "string",
      "searchTextCriteria": "string"
    },
    "batchId": "string",
    "multiDocumentSessionId": "string",
    "documentIntermediaryId": "string",
    "cctStatusCode": "string",
    "lastTimestampFrom": 0,
    "minCreationDate": "2021-02-24T23:34:28.076Z",
    "maxCreationDate": "2021-02-24T23:34:28.076Z",
    "expiryDate": "2021-02-24T23:34:28.076Z",
    "signerTextFieldsSearchCriteria": "string"
  }
}
```

Retrieve detail of single archived document

[GET] [/v1/archivedDocuments/{hubId}](#)

Permette di recuperare tutte le info di un determinato documento che è stato cancellato

Query params

- `hubId` identificativo del documento

TEMPLATES

Get templates list

[GET] [/v1/templates](#)

Permette di avere una lista dei template creati

Query params

- `page` numero di pagina
 - `size` numero di elementi per pagina
 - `sort` campo per cui ordinare i documenti
 - `ownerId` identificativo azienda
-

Retrieve detail of single template

[GET] </v1/templates/{templateId}>

Permette di recuperare tutte le info di un determinato template

Path params

- `templateId` id del template
-

Download base64 template document

[GET] </v1/templates/{templateId}/templateDocuments/{templateDocumentId}/download>

Permette il download in base64 di un determinato documento inserito nel template

Path params

- `templateId` id del template
 - `templateDocumentId` id del documento
-

Download template document

[GET]
</v1/templates/{templateId}/templateDocuments/{templateDocumentId}/download/file>

Permette il download del file di un determinato documento inserito nel template

Path params

- `templateId` id del template
- `templateDocumentId` id del documento

//TODO MUOVERE NELLA SEZIONE ESEMPI

Esempi di utilizzo

Fetch documenti incrementale

Per ottenere i documenti in maniera incrementale è necessario utilizzare l'endpoint **/v1/documents/search** specificando un filtro per **lastTimestampFrom** ed un ordinamento per **lastTimestamp**.

```
POST /api/v1/documents/search?page=0&size=100&sort=lastTimestamp,ASC

body:
{
  "lastTimestampFrom" : 1587736390388
}
```

Il timestamp è uno unix epoch in millisecondi e la prima volta che si effettua la chiamata non è necessario specificarlo.

Ad ogni chiamata è necessario salvarsi il valore più grande del campo **lastTimestamp** presente sui documenti ottenuti all'ultima pagina richiesta (se il sorting è ascendente, in caso di sorting discendente va preso il primo documento della prima pagina).

Di seguito un esempio:

```
--- chiamata 1 ---
POST /api/v1/documents/search?page=0&size=100&sort=lastTimestamp,ASC

body: { }
```

response:

```
{
  "_embedded": {
    "documentList": [
      ... documenti ...
    ]
  }
}
```

```

    ]
  },
  "_links": {
    []... links hateoas ...
  },
  "page": {
    "size": 100,
    "totalElements": 150,
    "totalPages": 2,
    "number": 0
  }
}

```

--- chiamata 2 ---

POST /api/v1/documents/search?page=1&size=100&sort=lastTimestamp,ASC

body: { }

response:

```

{
  "_embedded": {
    "documentList": [
      [] ... primi 49 documenti ...,
      {
        ... campi 50° documento ...,
        "lastTimestamp": 1587736390388
      }
    ]
  },
  "_links": {
    []... links hateoas ...
  },
  "page": {
    "size": 100,
    "totalElements": 150,
    "totalPages": 2,
    "number": 1
  }
}

```

Finito il primo giro di richieste paginate sarà necessario salvare il campo `lastTimestamp` più grande trovato nei documenti ottenuti, in questo esempio il valore **1587736390388**.

Alla richiesta successiva sarà necessario impostare il suddetto valore come filtro **lastTimestampFrom** nel body della richiesta:

```
--- chiamata 1 ---
POST /api/v1/documents/search?page=0&size=100&sort=lastTimestamp,ASC

body:
{
  "lastTimestampFrom": 1587736390388
}

response:
{
  "_embedded": {
    "documentList": [
      ... 9 documenti ...,
      {
        ... campi 10° documento ...,
        "lastTimestamp": <nuovo valore da salvare>
      }
    ]
  },
  "_links": {
    ... links hateoas ...
  },
  "page": {
    "size": 100,
    "totalElements": 10,
    "totalPages": 1,
    "number": 0
  }
}
```

In questo modo ad ogni nuova chiamata saranno presenti solo documenti che hanno subito una variazione, quindi con un `lastTimestamp` maggiore a quello ottenuto dalla chiamata precedente.