

Signature Write

API di scrittura 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

UPLOAD

Upload file for async sign

[POST] [/v1/rpc/upload/async](#)

Permette di caricare un documento per il [processo asincrono](#)

Request body

I campi contrassegnati da **asterisco (*)** sono da considerarsi **obbligatori**.

- `transmitterId` (*) id dell'azienda
- `managerId` id del commercialista dell'azienda
- `batchId`
- `fileName` (*) nome del file da caricare
- `fileContent` (*) base64 del file da caricare. il limite è di **15MB**
- `documentType` (*) tipo documento, consultare [qui](#) la lista dei tipi di documento disponibili
- `sessionName` nome personalizzato da assegnare alla sessione
- `notes` note aggiuntive sulla sessione
- `thirdPartyEmailAddresses` array di indirizzi email che riceveranno il documento firmato via mail
- `owner` (*) dati relativi all'intestatario del documento, può essere una **persona fisica** o **azienda**

- `owner.firstName`
- `owner.lastName`
- `owner.companyName` ragione sociale
- `owner.vat` partita iva
- `owner.fiscalCode` codice fiscale
- `owner.kind` tipo dell'intestatario. i valori possibili sono **company** o **person**
- `metadata` dati aggiuntivi per la conservazione del documento. **Chiedere info in più** perchè ogni tipo di documento ha dei metadati di conservazione diversi. Oltre ai metadati specifici per la singola classe documentale, per ogni classe documentale bisogna aggiungere questi campi obbligatori:
 - `SoggettoTitolare-CodiceFiscale`
 - `SoggettoTitolare-PartitaIva`
 - `SoggettoTitolare-Nome` e `SoggettoTitolare-Cognome` in caso di persona fisica altrimenti `SoggettoTitolare-Denominazione`
 - `SoggettoTitolare-IdPaese`
- `autoSendCCT` permette di mandare automaticamente il documento in conservazione una volta che il documento è stato firmato. Per far sì che la conservazione avvenga serve che l'azienda abbia il **servizio di cct attivo** e abbia inserito tutti i **metadata obbligatori** per lo specifico documento
- `signatureMode` (*) i valori che può assumere sono **pades** o **cados**
- `mainSignerId` identificativo di un firmatario che ha un certificato di **FEQ automatica** per sigillare il documento. Se si è scelto di utilizzare il **sigillo TS** bisogna impostarlo a **null**
- `expiryDate` (*) data di scadenza del documento, dopo 30 giorni dalla data di invio del documento alla firma per le sessioni **FEA** e **FES** e 60 giorni per le sessioni di **FEQ** il documento andrà automaticamente in stato scaduto se non verrà firmato
- `strictExpiryDate` se impostato a **true** ed è presente una data di scadenza il documento, se non verrà firmato entro quella data, passerà in stato **SCADUTO**. Se invece viene impostato a false non verrà fatto il check sulla data di scadenza inserita ma saranno validi i 30/60 giorni specificati sopra per la scadenza del documento
- `year` anno di riferimento del documento
- `signatureProcessName` (*) tipo di processo di firma. I valori possono essere:
 - **FEA remota**
 - **FEA grafometrica**
 - **FES remota**
 - **FES grafometrica**
 - **FEQ remota**
 - **FEQ con dispositivo fisico**
- `areSignersOrdered`: parametro che serve per specificare che i firmatari devono essere ordinati e devono firmare in modo consecutivo
- `attachments` eventuali allegati del documento. In questa fase, gli allegati verranno solo specificati mentre il caricamento verrà effettuato durante il processo di firma della sessione
- `attachments.description`: descrizione dell'allegato
- `attachments.signerFiscalCode`: codice fiscale del firmatario a cui richiedere l'allegato (deve essere presente tra i firmatari)

- `attachments.details`: array dei **file** che verranno richiesti nella sessione di firma. Indipendentemente dal numero di `attachments.details` specificati il risultato finale sarà un PDF con l'aggregazione dei details di un singolo allegato.
- `attachments.details.description`: descrizione del **file** da caricare
- `attachments.details.optional`: impostare a `true` se non si vuol rendere l'allegato obbligatorio

- `signers` (*) array di firmatari che devono firmare il documento
- `signers.fiscalCode`
- `signers.firstName`
- `signers.lastName`
- `signers.email` email del firmatario
- `signers.mobile` numero del **telefono cellulare** del firmatario, verrà utilizzato per la sessione di firma
- `signers.canEmailBeSentAgain` //TODO DA ELIMINARE

- `signers.signatureData` array delle area di firma del firmatario sul documento
- `signers.signatureData.signatureType` identifica il tipo di firma che verrà effettuata, **tutte le area di firma sul documento** devono avere lo stesso valore. I tipi disponibili sono:
 - **SecureCall** tipologia disponibile per **FEA remota** e **FES remota**
 - **otp** tipologia disponibile per **FEA remota** e **FES remota**
 - **BioSignature** tipologia che serve per la **FEA grafometrica in presenza**
 - **BioSignatureRemote** tipologia che serve per la **FEA grafometrica da remoto**
 - **fes_noservice** tipologia che serve per la **FES grafometrica in presenza**
 - **fes_noservice_remote** tipologia che serve per la **FES grafometrica da remoto**
 - **fes** tipologia che serve per la **FES grafometrica con ausilio di tavoletta grafometrica Wacom**
 - **FSM** tipologia che serve per la **FEQ remota**, il valore può essere oltre a FSM anche **FDS** se il certificato di firma qualificata remota è stato richiesto con il metodo di autenticazione **otp**
 - **FSC** tipologia che serve per la **FEQ Remota con dispositivo fisico**
- `signers.signatureData.fieldGroup` può essere valorizzato con un valore personalizzato. Se lo stesso **fieldGroup** verrà messo su più aree di firma il firmatario le firmerà contemporaneamente altrimenti verrà richiesta una firma per ogni singola aree di firma presente sul documento
- `signers.signatureData.shortDescription` descrizione corta dell'area sul documento. Limite caratteri tra 5 e 20
- `signers.signatureData.longDescription` descrizione lunga dell'area sul documento. Limite caratteri 100
- `signers.signatureData.optional` impostare a **true** se non si vuol rendere l'area di firma obbligatoria
- `signers.signatureData.textType` identifica il tipo di area sul documento. Se non impostato di default verrà messo come area di firma. **Il documento deve contenere almeno un area di firma per essere valido**. I tipi possono essere:
 - **signature** area di firma

- **checkbox** per inserire una checkbox, oltre al `textType` dovrà essere inserito come `longDescription` il titolo che si vuole avere la selezione o meno della checkbox
 - **radiobutton** per inserire un radio button, oltre al `textType` le varie scelte del radio button dovranno avere sia la `longDescription` che il `fieldGroup` uguale per ogni casella di scelta del radio button. La `shortDescription` identificherà le varie opzioni. Vedere esempio
 - **text** per inserire del testo, oltre al `textType` dovrà essere inserito come `longDescription` il titolo che si vuole avere per la sezione
 - `signers.signatureData.positioningStrategy` identifica il tipo di posizionamento che verrà usato per le varie aree sul documento. I valori possono essere:
 - **coords** vengono utilizzate delle coordinate per identificare il posizionamento
 - **textPattern** viene utilizzato un id all'interno del documento che farà da riferimento per il posizionamento
-
- `coordsRegion.page`
 - `coordsRegion.h`
 - `coordsRegion.w`
 - `coordsRegion.x`
 - `coordsRegion.y`
-
- `textPatternRegion.pattern` identificativo sul documento dell'area
 - `textPatternRegion.h`
 - `textPatternRegion.w`
 - `textPatternRegion.offsetX`
 - `textPatternRegion.offsetY`
-
- `multipleDocumentSession` se impostato a true il documento farà parte di una sessione multi documento (più documenti verranno firmati con un'unica sessione di firma)
 - `useSignatureTimestamp` permette di applicare una marca temporale al documento
 - `sendSignedDocumentToMail` se impostato a true i firmatari presenti nel documento alla fine della firma del documento riceveranno una mail con in allegato il documento firmato
 - `webHookUrl` permette di specificare un url dove verranno mandati gli eventi relativi all'aggiornamento di stato del documento.
Esempio del body del webhook che verrà mandato
 - `afterSignRedirectUrl` permette di specificare un url su cui l'utente verrà reindirizzato a processo di firma terminato.

```
{
  "hubId": "616ecd1edbd9bc7f725633da",
  "status": {
    "statusCode": "Firmato",
    "errorDescription": null,
    "debugDescription": null,
  }
}
```

```
    "creationDate": "2021-10-19T13:51:58.696Z"
  }
}
```

```
{
  "transmitterId": "string",
  "managerId": "string",
  "batchId": "string",
  "fileName": "string",
  "sessionName": "string",
  "notes": "string",
  "thirdPartyEmailAddresses": [
    "string",
    ...
    "string"
  ]
  "fileContent": "string",
  "documentType": "string",
  "owner": {
    "firstName": "string",
    "lastName": "string",
    "companyName": "string",
    "vat": "string",
    "fiscalCode": "string",
    "kind": "string"
  },
  "metadata": {
    "cct": {
      "additionalProp1": "string",
      "additionalProp2": "string",
      "additionalProp3": "string"
    },
    "document": {
      "additionalProp1": "string",
      "additionalProp2": "string",
      "additionalProp3": "string"
    }
  },
}
```

```
"autoSendCCT": true,
"signatureMode": "string",
"mainSignerId": "string",
"expiryDate": "string",
"year": 0,
"signatureProcessName": "string",
"areSignersOrdered": true,
"webHookUrl": "string",
"afterSignRedirectUrl": "string",
"attachments": [
  {
    "description": "string",
    "signerFiscalCode": "string",
    "details": [
      {
        "description": "string",
        "optional": true
      }
    ]
  }
],
"signers": [
  {
    "fiscalCode": "string",
    "firstName": "string",
    "lastName": "string",
    "email": "string",
    "mobile": "string",
    "canEmailBeSentAgain": true,
    "signatureData": [
      {
        "signatureType": "string",
        "fieldGroup": "string",
        "shortDescription": "string",
        "longDescription": "string",
        "textType": "string",
        "positioningStrategy": "string",
        "coordsRegion": {
          "page": 0,
          "x": 0,
```

```

        "y": 0,
        "w": 0,
        "h": 0
    },
    "fieldNameRegion": {
        "name": "string",
        "offsetX": 0,
        "offsetY": 0,
        "w": 0,
        "h": 0
    },
    "textPatternRegion": {
        "pattern": "string",
        "number": 0,
        "offsetX": 0,
        "offsetY": 0,
        "w": 0,
        "h": 0
    },
    "optional": true
}
]
}
],
"commonSign": true,
"multipleDocumentSession": true,
"useSignatureTimestamp": true,
"sendSignedDocumentToMail": true
}

```

[Esempi](#)

Upload file and download signed document

[POST] [/v1/rpc/upload/sync](#)

Permette il caricamento di un file seguendo il [processo sincrono](#)

Request body

- `transmitterId` id dell'azienda
- `managerId` id del commercialista dell'azienda
- `batchId`
- `fileName` nome del file da caricare
- `fileContent` base64 del file da caricare. il limite è di **30MB**
- `documentType` tipo documento, consultare [qui](#) la lista dei tipi disponibili
- `owner` dati relativi all'intestatario del documento, può essere una **persona fisica o azienda**
 - `owner.firstName`
 - `owner.lastName`
 - `owner.companyName` ragione sociale
 - `owner.vat` partita iva
 - `owner.fiscalCode` codice fiscale
 - `owner.kind` tipo dell'intestatario. i valori possibili sono **company** o **person**
 - `metadata` dati aggiuntivi per la conservazione del documento. **Chiedere info in più** perchè ogni tipo di documento ha dei metadati di conservazione diversi. Oltre ai metadati specifici per la singola classe documentale per ogni classe documentale bisogna aggiungere questi campi obbligatori:
 - `SoggettoTitolare-CodiceFiscale`
 - `SoggettoTitolare-PartitaIva`
 - `SoggettoTitolare-Nome` e `SoggettoTitolare-Cognome` in caso di persona fisica sennò `SoggettoTitolare-Denominazione`
 - `SoggettoTitolare-IdPaese`
- `autoSendCCT` permette di mandare automaticamente il documento in conservazione una volta che il documento è stato firmato, per far si che la conservazione avvenga serve che l'azienda abbia il **servizio di cct attivo** e abbia inserito tutti i **metadata obbligatori** per lo specifico documento
- `signatureMode` i valori che può assumere sono **pades** o **cados** o **xades**
- `signer` firmatario con un certificato di **FEQ automatica**
 - `signer.fiscalCode`
 - `signer.firstName`
 - `signer.lastName`
 - `signer.email` email del firmatario
 - `signer.mobile` numero del **telefono cellulare** del firmatario
 - `signer.canEmailBeSentAgain` //TODO DA ELIMINARE
 - `pin` deprecato
 - `useSignatureTimestamp` permette di applicare una marca temporale al documento
 - `withoutFile` se impostato a true nella response non verrà riportato il base64 del file firmato

```
{
  "transmitterId": "string",
  "managerId": "string",
```



```
"batchId": "string",
"fileName": "string",
"fileContent": "string",
"documentType": "string",
"owner": {
  "firstName": "string",
  "lastName": "string",
  "companyName": "string",
  "vat": "string",
  "fiscalCode": "string",
  "kind": "string"
},
"metadata": {
  "cct": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "document": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  }
},
"autoSendCCT": true,
"signatureMode": "string",
"signer": {
  "fiscalCode": "string",
  "firstName": "string",
  "lastName": "string",
  "email": "string",
  "mobile": "string",
  "canEmailBeSentAgain": true,
  "pin": "string"
},
"withoutFile": true,
"useSignatureTimestamp": false
}
```

V2 - UPLOAD

Upload file for async sign

[POST] [/v2/rpc/upload/async](#)

Permette di caricare un documento per il [processo asincrono](#)

Request body

- `transmitterId` id dell'azienda
- `managerId` id del commercialista dell'azienda
- `batchId`
- `fileName` nome del file da caricare
- `fileContent` base64 del file da caricare. il limite è di **15MB**
- `documentType` tipo documento, consultare [qui](#) la lista dei tipi disponibili
- `sessionName` nome personalizzato da assegnare alla sessione
- `notes` note aggiuntive sulla sessione
- `owner` dati relativi all'intestatario del documento, può essere una **persona fisica o azienda**
 - `owner.firstName`
 - `owner.lastName`
 - `owner.companyName` ragione sociale
 - `owner.vat` partita iva
 - `owner.fiscalCode` codice fiscale
 - `owner.kind` tipo dell'intestatario. i valori possibili sono **company** o **person**
 - `metadata` dati aggiuntivi per la conservazione del documento. **Chiedere info in più** perchè ogni tipo di documento ha dei metadati di conservazione diversi. Oltre ai metadati specifici per la singola classe documentale per ogni classe documentale bisogna aggiungere questi campi obbligatori:
 - `SoggettoTitolare-CodiceFiscale`
 - `SoggettoTitolare-PartitaIva`
 - `SoggettoTitolare-Nome` e `SoggettoTitolare-Cognome` in caso di persona fisica sennò `SoggettoTitolare-Denominazione`
 - `SoggettoTitolare-IdPaese`
- `autoSendCCT` permette di mandare automaticamente il documento in conservazione una volta che il documento è stato firmato, per far sì che la conservazione avvenga serve che l'azienda abbia il **servizio di cct attivo** e abbia inserito tutti i **metadata obbligatori** per lo specifico documento
- `signatureMode` i valori che può assumere sono **pades** o **cades**
- `mainSignerId` identificativo di un firmatario che ha un certificato di **FEQ automatica** per sigillare il documento. Se si scelto di utilizzare il **sigillo TS** bisogna impostarlo a **null**
- `expiryDate` data di scadenza del documento, dopo 30 giorno dalla data di invio del documento alla firma per le sessioni **FEA** e **FES** e 60 giorni per le sessioni di **FEQ** il documento andrà automaticamente in stato scaduto se non verrà firmato

- `strictExpiryDate` se impostato a **true** ed è presente una data di scadenza il documento se non verrà firmato entro quella data passerà in stato **SCADUTO** se invece viene impostato a false non verrà fatto il check sulla data di scadenza inserita ma saranno validi i 30/60 giorni specificati sopra per la scadenza del documento
 - `year` anno di riferimento del documento
 - `signatureProcessName` tipo di processo di firma. I valori possono essere:
 - **FEA remota**
 - **FEA grafometrica**
 - **FES remota**
 - **FES grafometrica**
 - **FEQ remota**
 - **FEQ con dispositivo fisico**
 - `areSignersOrdered`: parametro che serve per specificare che i firmatari devono essere ordinati e devono firmare in modo consecutivo
 - `attachments` eventuali allegati del documento, gli allegati qui verranno solo specificati mentre il caricamento verrà effettuato durante il processo di firma della sessione
 - `attachments.description`: descrizione dell'allegato
 - `attachments.signerFiscalCode`: codice fiscale del firmatario a cui richiedere l'allegato (deve essere presente tra i firmatari)
 - `attachments.details`: array dei **file** che verranno richiesti nella sessione di firma. Indipendentemente dal numero di `attachments.details` specificati il risultato finale sarà un PDF con l'aggregazione dei details di un singolo allegato.
 - `attachments.details.description`: descrizione del **file** da caricare
 - `attachments.details.optional`: impostare a true se non si vuol rendere l'allegato obbligatorio
-
- `signers` array di firmatari che devono firmare il documento
 - `signers.fiscalCode`
 - `signers.firstName`
 - `signers.lastName`
 - `signers.email` email del firmatario
 - `signers.mobile` numero del **telefono cellulare** del firmatario, verrà utilizzato per la sessione di firma
 - `signers.enrollmentInfo` oggetto che consente di passare le informazioni di un firmatario al quale si vuole attivare un certificato di firma qualificata per potergli poi far firmare un documento, al quale devono essere passate le seguenti property:
 - `signerId` identificativo del firmatario.
 - `profile` che può avere uno dei questi determinati valori:
 - **"teamsystemTriennale"**
 - **"teamsystemAuto"**
 - **"teamsystemOneshot"**
 - `identificationProcesses` array che definisce la modalità di riconoscimento, i valori ammessi sono:
 - **"FEQ"**
 - **"SPID"**

- `enrollmentOwner` informazioni del firmatario.
 - `enrollmentOwner.firstname`
 - `enrollmentOwner.lastName`
 - `enrollmentOwner.email`
 - `enrollmentOwner.phoneNumber`
 - `enrollmentOwner.fiscalCode` (facoltativo)
-
- `signers.signatureData` array delle aree di firma del firmatario sul documento
 - `signers.signatureData.signatureType` identifica il tipo di firma che verrà effettuata, **tutte le aree di firma sul documento** devono avere lo stesso valore. I tipi disponibili sono:
 - **SecureCall** tipologia disponibile per **FEA remota** e **FES remota**
 - **otp** tipologia disponibile per **FEA remota** e **FES remota**
 - **BioSignature** tipologia che serve per la **FEA grafometrica in presenza**
 - **BioSignatureRemote** tipologia che serve per la **FEA grafometrica da remoto**
 - **fes_noservice** tipologia che serve per la **FES grafometrica in presenza**
 - **fes_noservice_remote** tipologia che serve per la **FES grafometrica da remoto**
 - **fes** tipologia che serve per la **FES grafometrica con ausilio di tavoletta grafometrica Wacom**
 - **FSM** tipologia che serve per la **FEQ remota**, il valore può essere oltre a FSM anche **FDS** se il certificato di firma qualificata remota è stato richiesto con il metodo di autenticazione **otp**
 - **FSC** tipologia che serve per la **FEQ Remota con dispositivo fisico**
 - `signers.signatureData.fieldGroup` può essere valorizzato con un valore personalizzato, se lo stesso **fieldGroup** verrà messo su più aree di firma il firmatario le firmerà contemporaneamente senno verrà richiesta una firma per ogni singola aree di firma presente sul documento
 - `signers.signatureData.shortDescription` descrizione corta dell'area sul documento Limite caratteri tra 5 e 20
 - `signers.signatureData.longDescription` descrizione lunga dell'area sul documento. Limite caratteri 100
 - `signers.signatureData.optional` impostare a **true** se non si vuol rendere l'area di firma obbligatoria
 - `signers.signatureData.textType` identifica il tipo di area sul documento, se non impostato di default verrà messo come area di firma. Il documento deve contenere almeno un area di firma per essere valido. I tipi possono essere:
 - **signature** area di firma
 - **checkbox** per inserire una checkbox, oltre al `textType` dovrà essere inserito come `longDescription` il titolo che si vuole avere la selezione o meno della checkbox
 - **radiobutton** per inserire un radio button, oltre al `textType` le varie scelte del radio button dovranno avere sia la `longDescription` che il `fieldGroup` uguale per ogni casella di scelta del radio button. La `shortDescription` identificherà le varie opzioni. Vedere esempio
 - **text** per inserire del testo, oltre al `textType` dovrà essere inserito come `longDescription` il titolo che si vuole avere per la sezione
 - `signers.signatureData.positioningStrategy` identifica il tipo di posizionamento che verrà usato per le varie aree sul documento. I valori possono essere:

- **coords** vengono utilizzate delle coordinate per identificare il posizionamento
- **textPattern** viene utilizzato un id all'interno del documento che farà da riferimento per il posizionamento

- `coordsRegion.page`
- `coordsRegion.h`
- `coordsRegion.w`
- `coordsRegion.x`
- `coordsRegion.y`

- `textPatternRegion.pattern` identificativo sul documento dell'area
- `textPatternRegion.h`
- `textPatternRegion.w`
- `textPatternRegion.offsetX`
- `textPatternRegion.offsetY`

- `multipleDocumentSession` se impostato a true il documento farà parte di una sessione multi documento (più documenti verranno firmati con un'unica sessione di firma)
- `useSignatureTimestamp` permette di applicare una marca temporale al documento
- `sendSignedDocumentToMail` se impostato a true i firmatari presenti nel documento alla fine della firma del documento riceveranno una mail con in allegato il documento firmato
- `webHookUrl` permette di specificare un url dove verranno mandati gli eventi relativi all'aggiornamento di stato del documento.
- `afterSignRedirectUrl` permette di specificare un url su cui l'utente verrà reindirizzato a processo di firma terminato.

Esempio del body del webhook che verrà mandato

```
{
  "hubId": "616ecd1edbd9bc7f725633da",
  "status": {
    "statusCode": "Firmato",
    "errorDescription": null,
    "debugDescription": null,
    "creationDate": "2021-10-19T13:51:58.696Z"
  }
}
```

```
{
  "transmitterId": "string",
  "managerId": "string",
```

```
"batchId": "string",
"fileName": "string",
"fileContent": "string",
"documentType": "string",
"sessionName": "string",
"notes": "string",
"owner": {
  "firstName": "string",
  "lastName": "string",
  "companyName": "string",
  "vat": "string",
  "fiscalCode": "string",
  "kind": "string"
},
"metadata": {
  "cct": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "document": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  }
},
"autoSendCCT": true,
"signatureMode": "string",
"mainSignerId": "string",
"expiryDate": "string",
"year": 0,
"signatureProcessName": "string",
"areSignersOrdered": true,
"webHookUrl": "string",
"afterSignRedirectUrl": "string",
"attachments": [
  {
    "description": "string",
    "signerFiscalCode": "string",
    "details": [
```

```
{
  "description": "string",
  "optional": true
}
]
}
],
"signers": [
  {
    "fiscalCode": "string",
    "firstName": "string",
    "lastName": "string",
    "email": "string",
    "mobile": "string",
    "canEmailBeSentAgain": true,
    "signatureData": [
      {
        "signatureType": "string",
        "fieldGroup": "string",
        "shortDescription": "string",
        "longDescription": "string",
        "textType": "string",
        "positioningStrategy": "string",
        "coordsRegion": {
          "page": 0,
          "x": 0,
          "y": 0,
          "w": 0,
          "h": 0
        },
        "fieldNameRegion": {
          "name": "string",
          "offsetX": 0,
          "offsetY": 0,
          "w": 0,
          "h": 0
        },
        "textPatternRegion": {
          "pattern": "string",
          "number": 0,
```

```

        "offsetX": 0,
        "offsetY": 0,
        "w": 0,
        "h": 0
    },
    "optional": true
}
]
}
],
"commonSign": true,
"multipleDocumentSession": true,
"useSignatureTimestamp": true,
"sendSignedDocumentToMail": true
}

```

Esempi

Upload file and download signed document

[POST] [/v2/rpc/upload/sync](#)

Permette il caricamento di un file seguendo il [processo sincrono](#)

Request body

- `transmitterId` id dell'azienda
- `managerId` id del commercialista dell'azienda
- `batchId`
- `fileName` nome del file da caricare
- `fileContent` base64 del file da caricare. il limite è di **30MB**
- `documentType` tipo documento, consultare [qui](#) la lista dei tipi disponibili
- `owner` dati relativi all'intestatario del documento, può essere una **persona fisica** o **azienda**
 - `owner.firstName`
 - `owner.lastName`
 - `owner.companyName` ragione sociale
 - `owner.vat` partita iva
 - `owner.fiscalCode` codice fiscale
 - `owner.kind` tipo dell'intestatario. i valori possibili sono **company** o **person**

- `metadata` dati aggiuntivi per la conservazione del documento. **Chiedere info in più** perchè ogni tipo di documento ha dei metadati di conservazione diversi. Oltre ai metadati specifici per la singola classe documentale per ogni classe documentale bisogna aggiungere questi campi obbligatori:
 - `SoggettoTitolare-CodiceFiscale`
 - `SoggettoTitolare-PartitaIva`
 - `SoggettoTitolare-Nome` e `SoggettoTitolare-Cognome` in caso di persona fisica senno `SoggettoTitolare-Denominazione`
 - `SoggettoTitolare-IdPaese`
- `autoSendCCT` permette di mandare automaticamente il documento in conservazione una volta che il documento è stato firmato, per far si che la conservazione avvenga serve che l'azienda abbia il **servizio di cct attivo** e abbia inserito tutti i **metadata obbligatori** per lo specifico documento
- `signatureMode` i valori che può assumere sono **pades** o **cados** o **xades**
- `signer` firmatario con un certificato di **FEQ automatica**
- `signer.fiscalCode`
- `signer.firstName`
- `signer.lastName`
- `signer.email` email del firmatario
- `signer.mobile` numero del **telefono cellulare** del firmatario
- `signer.canEmailBeSentAgain` //TODO DA ELIMINARE
- `pin` deprecato
- `withoutFile` se impostato a true nella response non verrà riportato il base64 del file firmato

```
{
  "transmitterId": "string",
  "managerId": "string",
  "batchId": "string",
  "fileName": "string",
  "fileContent": "string",
  "documentType": "string",
  "owner": {
    "firstName": "string",
    "lastName": "string",
    "companyName": "string",
    "vat": "string",
    "fiscalCode": "string",
    "kind": "string"
  },
  "metadata": {
    "cct": {
```

```
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "document": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  }
},
"autoSendCCT": true,
"signatureMode": "string",
"signer": {
  "fiscalCode": "string",
  "firstName": "string",
  "lastName": "string",
  "email": "string",
  "mobile": "string",
  "canEmailBeSentAgain": true,
  "pin": "string"
},
"withoutFile": true
}
```

Append document to multiDocumentSession

Desc

JOB

Start signature job

[POST] </v1/jobs/start>

Permette di creare la sessione di firma del [processo asincrono](#) per firmare il documento

Request body

- `hubId` identificativo del documento
- `sendEmail` **default = true**, permette di ricevere una mail con il link per la sessione di firma per ogni firmatario

[Esempio](#)

Start signature job for multiDocuments

[POST] [/v1/jobs/start/multidocument](#)

Permette di creare la sessione di firma del [processo asincrono](#) contemporaneamente su più documenti(documenti multi sessione)

Request body

- `multiDocumentSessionId` id del documento multi sessione
- `sendEmail` **default = true**, permette di ricevere una mail con il link per la sessione di firma per ogni firmatario

[Esempio](#)

V2 - JOB

Start signature job per CA Teamsystem

[POST] [/v2/jobs/start](#)

Permette di creare la sessione di firma del [processo asincrono](#) per firmare il documento

Request body

- `hubId` identificativo del documento
- `sendEmail` **default = true**, permette di ricevere una mail con il link per la sessione di firma per ogni firmatario

Esempio

Start signature job for multiDocuments

[POST] </v2/jobs/start/multidocument>

Permette di creare la sessione di firma del [processo asincrono](#) contemporaneamente su più documenti(documenti multi sessione)

Request body

- `multiDocumentSessionId` id del documento multi sessione
- `sendEmail` **default = true**, permette di ricevere una mail con il link per la sessione di firma per ogni firmatario

Esempio

Cancel signature job

[POST] </v2/jobs/cancel>

Permette di cancellare una sessione di firma

Request body

- `hubId` id del documento in sessione
- `multiDocumentSessionId` id della multi sessione

CCT

Send document to cct

[POST] </v1/cct/send>

Permette di inviare un documento in conservazione

Request body

- `hubId` identificativo del documento

Send documents to cct

[POST] [/v1/cct/send/multi](#)

Permette di inviare più documenti in conservazione per un massimo di 50 documenti per chiamata

Request body

- `hubIds` array di hubId

MAIL

Resend signature mail

[POST] [/v1/mail/resend](#)

Permette di rimandare la mail con il link per la sessione di firma per uno o più firmatari di un documento se non hanno ancora firmato

Request body

- `hubId` identificativo del documento
- `fiscalCodes` array dei codici fiscali dei firmatari presenti all'interno del documento che devono ricevere di nuovo la mail

DOCUMENT

Archive a document

[POST] [/v1/document/{hubId}/archive](#)

Permette di eliminare un documento. Il documento verrà spostato in una collection quindi per la ricerca eventuale del documento eliminato bisogna utilizzare [l'api apposita](#)

Path params

- `hubId` identificativo del documento
-

Archive a multiDocument session

[POST] </v1/document/multiDocumentSession/{multiDocumentSessionId}/archive>

Permette di eliminare un documento multi sessione. Il documento verrà spostato in una collection quindi per la ricerca eventuale del documento eliminato bisogna utilizzare [l'api apposta](#)

Path params

- `multiDocumentSessionId` id del documento multi sessione

Unarchive a document

[POST] </v1/document/{hubId}/unarchive>

Permette di ripristinare un documento eliminato. Solamente i documenti in stato Firmato o Caricato possono essere ripristinati

Path params

- `hubId` identificativo del documento

Unarchive a multiDocument session

[POST] </v1/document/multiDocumentSession/{multiDocumentSessionId}/unarchive>

Permette di ripristinare un documento multi sessione eliminato. Solamente i documenti in stato **Firmato** o **Caricato** possono essere ripristinati

Path params

- `multiDocumentSessionId` id del documento multi sessione

Remove from multiDocument session

[PATCH] </v1/document/{hubId}/removeFromMultiDocumentSession>

Permette di eliminare un documento da una sessione multi documento. Il documento nello specifico non verrà cancellato ma verrà considerato con un normale singolo documento //TODO

Path params

- `hubId` identificativo del documento

TEMPLATE

Titolo

Desc

Titolo

Desc

Titolo

Desc

Titolo

Desc

Titolo

Desc

Titolo

Desc

Cancel Jobs

POST /v1/jobs/cancel

Permette di cancellare un job precedentemente avviato. Questo comporta l'eliminazione o lo stralcio del documento. Lo stato del documento passerà in **ELIMINATO** se tutto il processo di firma non era ancora concluso altrimenti in **STRALCIATO**. I documenti già mandati in CCT non possono essere ne eliminati ne stralciati.

Revision #41

Created 2021-02-26 00:41:07 UTC by Raffaele Mainini

Updated 2024-12-19 12:10:37 UTC by Paolo Desantis