

API

- [Introduzione](#)
- [Incasso](#)
- [Conti](#)
- [Pagamento](#)
- [Consenso SCA](#)
- [Banche e Prodotti](#)
- [Settings](#)

Introduzione

Swagger

- [DEV](#)
- [TEST](#)

Headers

Per chiamare correttamente gli endpoint messi a disposizione dal servizio TS Pay Gateway, è necessario includere gli headers come descritto [qui](#).

Struttura dell'endpoint

Gli endpoint sono costruiti secondo la struttura:

```
/api/v{X}/{itemId}/...
```

dove

- `vX` dove X è la versione dell'API (al momento solo **v1**)
- `itemId` è il company registry dell'azienda per la quale si richiede il servizio

In questa guida tutti gli endpoint sono descritti senza il prefisso `/api/vX` che va dunque anteposto

Incasso

Fattura attiva

Descrizione

Consente di ricevere un pagamento attraverso il servizio di TS Pay; chiamando l'endpoint **link2pay** si riceverà un'url col quale sarà possibile fare un pagamento.

Richiesta link per effettuare un pagamento

Genera un link per effettuare un pagamento attraverso il servizio di TS Pay.

```
[POST] /{itemId}/link2pay
```

Nella **request** in particolare abbiamo:

- `template` contenente il testo ed eventualmente il logo da visualizzare nella pagina del pagamento generata e raggiungibile al link restituito in risposta
 - `title` titolo, generalmente usato per indicare la ragione sociale del merchant (es: "TeamSystem S.p.A.")
 - `desc` motivazione del pagamento (es: "Fattura di vendita")
 - `paymentRef` estremi della fattura (es: "Numero 123 del 15/06/2020")
 - `logo` link all'immagine da visualizzare come logo
- `callbackUrl` url cui reindirizzare l'utente una volta completata l'operazione. facoltativo
- `sourceTypes` sorgenti di pagamento accettate: carta di credito, SDD o entrambi (default)
- `maxPaymentsNumber` numero massimo di pagamenti consentiti (0=illimitato)
- `amount` l'importo da pagare espresso come numero intero in cui le ultime due cifre compongono la parte decimale (ad es. 1450 è da considerare come 14,50)
- `externalRef` un riferimento univoco al pagatore, da utilizzare in seguito come elemento di ricerca
- `metadata` elementi di tracciabilità, da utilizzare per la riconciliazione

Nella **response** l'API fornisce:

- `orderKey` identificativo dell'ordine
 - `url` link da utilizzare per eseguire l'operazione
-

Richiesta link per memorizzare una sorgente di pagamento per addebiti futuri

Genera un link per salvare una sorgente di pagamento attraverso il servizio di TS Pay.

```
[POST] /{itemId}/link2save
```

Nella **request** in particolare abbiamo:

- `template` contenente il testo ed eventualmente il logo da visualizzare nella pagina del pagamento generata e raggiungibile al link restituito in risposta
 - `title` titolo, generalmente usato per indicare la ragione sociale del merchant (es: "TeamSystem S.p.A.")
 - `desc` motivazione del pagamento (es: "Fattura di vendita")
 - `paymentRef` estremi della fattura (es: "Numero 123 del 15/06/2020")
 - `logo` link all'immagine da visualizzare come logo
- `callbackUrl` url cui reindirizzare l'utente una volta completata l'operazione. facoltativo
- `sourceTypes` sorgenti di pagamento accettate: carta di credito, SDD o entrambi (default)
- `maxSourceNumber` numero massimo di sorgenti memorizzabili (0=illimitato, default=1)
- `maxAmount` l'importo massimo per gli addebiti, inteso come cumulativo, espresso come numero intero in cui le ultime due cifre compongono la parte decimale (ad es. 1450 è da considerare come 14,50)
- `contextId` il contesto a cui la memorizzazione della sorgente si applica, quindi nel caso specifico la fattura
- `externalRef` un riferimento univoco al pagatore, da utilizzare in seguito come elemento di ricerca
- `metadata` elementi di tracciabilità, da utilizzare per la riconciliazione

Nella **response** l'API fornisce:

- `orderKey` identificativo dell'ordine
- `url` link da utilizzare per eseguire l'operazione

Addebito

Esegue l'addebito sulla sorgente salvata con il LinkToSave.

```
[POST] /{itemId}/charges
```

Tra gli **header** è previsto:

- `x-operation-key` chiave univoca dell'addebito con lo scopo di prevenire i doppi incassi, in modo che se si tenta una seconda operazione con la stessa chiave viene restituito un errore (opzionale)

Nella **request** in particolare abbiamo:

- `sourceKey` identifica la sorgente di pagamento memorizzata dal customer attraverso il `LinkToSave`
- `contextId` l'eventuale contesto precedentemente impostato
- `description` è la motivazione del pagamento, che finisce nel movimento
- `amount` l'importo da addebitare espresso come numero intero in cui le ultime due cifre compongono la parte decimale (ad es. 1450 è da considerare come 14,50)

Nella **response** l'API fornisce:

- `orderKey` identificativo dell'ordine
 - `chargeKey` identificativo dell'addebito
-

Esito pagamento

Da utilizzare per conoscere tutti i dettagli dell'addebito.

```
[GET] /{itemId}/charges/orders/{orderKey}
```

I **parametri** richiesti sono:

- `orderKey` path variable - key dell'ordine, ritornato dalla [chiamata alla creazione della richiesta di pagamento](#).

Conti

Aggregazione conti e carte

Descrizione

Consente di recuperare le informazioni aggregate di saldo e movimenti relativi a conti corrente e carte di credito tramite il servizio di TS Pay. La SCA è unica ma va rinnovata periodicamente, in genere ogni 90 giorni.

Informazioni conti collegati

Restituisce l'elenco dei prodotti (conti e carte) collegati, comprensivo di saldo disponibile e informazioni relative al consenso.

```
[GET] /{itemId}/linked-accounts
```

Nella **response** l'API fornisce:

- `accountId` id del conto/carta
- `iban` iban del conto (se il prodotto è un conto corrente)
- `maskedPan` numero della carta mascherato (se il prodotto è una carta)
- `currency` valuta di conto
- `providerId` id del provider (ASPSP)
- `bankName` nome della banca
- `productCode` codice del prodotto
- `lastBalance` saldo
 - `amount` importo (espresso come numero intero in cui le ultime due cifre compongono la parte decimale)
 - `currency` valuta
- `lastBalanceDate` data di riferimento del saldo
- `accountNature` natura del conto
- `consentId` id del consenso
- `consentExpireDate` data di scadenza del consenso
- `lastRefreshStatus` stato ultimo aggiornamento
- `lastRefreshDate` data ultimo aggiornamento

Informazioni conti collegati o precedentemente utilizzati

Restituisce l'elenco dei prodotti (solo conti) collegati o precedentemente utilizzati. I primi comprensivi di saldo disponibile e informazioni relative al consenso.

```
[GET] /{itemId}/accounts
```

Nella **response** l'API fornisce:

- `accountId` id del conto/carta (solo per conti collegati)
- `iban` iban del conto
- `currency` valuta di conto
- `providerId` id del provider (ASPSP)
- `bankName` nome della banca
- `productCode` codice del prodotto
- `lastBalance` saldo (solo per conti collegati)
 - `amount` importo (espresso come numero intero in cui le ultime due cifre compongono la parte decimale)
 - `currency` valuta
- `lastBalanceDate` data di riferimento del saldo (solo per conti collegati)

Informazioni movimenti

Restituisce le informazioni relative alle transazioni dei conti collegati.

```
[GET] /{itemId}/transactions
```

I **parametri** richiesti sono:

- `dateFrom` data inizio movimenti (opzionale)
- `dateTo` data fine movimenti (opzionale)
- `bookingStatus` stato del movimento (booked o pending) (opzionale)
- `accounts` accounts per i quali mostrare i movimenti (opzionale)

Nella **response** l'API fornisce:

- `accountId` identifica il conto a cui la transazione fa riferimento
- `bookingStatus` stato del movimento (booked o pending)
- `bookingDate` data del movimento
- `remittanceInformationUnstructured` descrizione del movimento
- `transactionAmount` importo movimento
 - `amount` importo (espresso come numero intero in cui le ultime due cifre compongono la parte decimale)

- `currency` valuta
 - `valueDate` data valuta
-

Aggiornamento dati di un conto

Una volta collegato un conto/carta è necessario fare un aggiornamento dei dati, per scaricare transazioni e saldi. Questa operazione è schedulata tutte le notti, ma si può anche eseguire a richiesta tramite la seguente API. Va fatta per tutti gli account collegati con quel determinato consenso.

```
[POST] /{itemId}/accounts/{accountId}/refresh
```

I **parametri** richiesti sono:

- `accountId` path variable - identifica il conto

Nel corpo della **request** in particolare abbiamo:

- `psuIpAddress` IP del client che effettua la richiesta (è richiesto un indirizzo IP valido)

Nella **response** l'API fornisce:

- `accountId` identifica il conto
- `status` stato dell'operazione di refresh (tipicamente *"updating"*)

Pagamento

Fattura passiva

Descrizione

Consente di effettuare un bonifico singolo, per una o più fatture/scadenze relative allo stesso beneficiario. La SCA è richiesta ad ogni operazione, fatte salve specifiche esenzioni previste dalla normativa.

Inizializzazione pagamento

Inizializza un pagamento tramite SCT.

```
[POST] /{itemId}/payments
```

Nella **request** in particolare abbiamo:

- `externalRef` un riferimento univoco al creditore, da utilizzare in seguito come elemento di ricerca
- `metadata` elementi di tracciabilità, da utilizzare per la riconciliazione
- `providerId` id del provider (ASPSP, la banca)
- `productCode` id del prodotto selezionato
- `psuIpAddress` indirizzo IP del richiedente
- `debtorAccount` conto del pagatore
 - `iban` iban
 - `currency` valuta
- `creditorAccount` conto del creditore
 - `iban` iban
 - `currency` valuta
 - `creditorName` nome
- `amount` importo (espresso come numero intero in cui le ultime due cifre compongono la parte decimale)
- `currency` valuta
- `requestedExecutionDate` data di esecuzione
- `requestedExecutionTime` ora di esecuzione
- `remittanceInformation` descrizione (causale)
- `paymentProduct` SCT o SCTinst
- `tppRedirectUri` URL di redirect

Per conoscere `productCode` e `providerId` va preventivamente invocato [l'endpoint per la lettura dei conti](#).

Nella **response** l'API fornisce:

- `paymentId` id del pagamento (al momento ciascun gruppo contiene un solo pagamento)
 - `status` stato dell'operazione
 - `totalTransactionFees` commissioni totali della transazione
 - `scaManagerRedirectUrl` URL di redirect allo SCA manager
-

Esito pagamento

Interroga l'esito del pagamento.

```
[GET] /{itemId}/payments/{paymentId}
```

I **parametri** richiesti sono:

- `paymentId` path variable - id del pagamento, ritornato dalla [chiamata all'inizializzazione dello stesso](#)
- `psuIpAddress` query string - indirizzo IP del client

Nella **response** l'API fornisce:

- `paymentId` id del pagamento
- `status` stato dell'operazione
- `totalTransactionFees` commissioni totali della transazione
- `totalAmount` importo totale della transazione
 - `amount` importo (espresso come numero intero in cui le ultime due cifre compongono la parte decimale)
 - `currency` valuta
- `debtorAccount` iban del debitore
- `creditorAccount` iban del creditore
- `createdOn` data operazione

Consenso SCA

Effettuare una SCA

Descrizione

In questa sezione vengono elencati gli endpoint che consentono di ottenere e rinnovare i vari consensi necessari al collegamento di un conto.

Ottenimento consenso una-tantum per mostrare la lista dei conti

Consente di effettuare la SCA (usa e getta) per accedere ai conti/carte in proprio possesso presso la banca, per il prodotto definito nel productId.

```
[POST] /{itemId}/consents/accountsList
```

Nella **request** in particolare abbiamo:

- `providerId*` id della banca (o provider)
- `productCode*` codice del prodotto
- `accountNature*` natura dell'account ("account" o "card")
- `tppRedirectUri` URL di redirect dopo che la SCA viene effettuata

Nella **response** l'API fornisce:

- `consentId` id del consenso
 - `consentStatus` stato del consenso
 - `scaManagerRedirectUrl` URI per effettuare la SCA
 - `validUntil` data fine validità
-

Ottenimento consenso ricorrente

Consente di effettuare la SCA (valida tipicamente 90 giorni) per collegare ai conti/carte in proprio possesso presso la banca, per il prodotto selezionato.

```
[POST] /{itemId}/consents/recurring
```

Nella **request** in particolare abbiamo:

- `providerId*` id della banca (o provider)
- `productCode*` codice del prodotto
- `tpRedirectUri` URL di redirect dopo che la SCA viene effettuata
- `accountAccess*` oggetto composto da
 - `balances*` lista di oggetti con la seguente forma:
 - `resourceId*`
 - `iban`
 - `maskedPan`
 - `currency*`
 - `accountNature`
 - `transactions*` lista di oggetti con la stessa forma di `balances`
 - `accounts` lista di oggetti con la stessa forma di `balances`

Nella **response** l'API fornisce:

- `consentId` id del consenso
- `consentStatus` stato del consenso
- `scaManagerRedirectUrl` URI per effettuare la SCA
- `validUntil` data fine validità

Rinnovo di un consenso

Consente di rinnovare la SCA (valida tipicamente 90 giorni) a partire da un consenso precedentemente valido.

```
[PUT] /{itemId}/consents/{consentId}/renew
```

I **parametri** richiesti sono:

- `consentId` path variable - id del consenso.

Nella **response** l'API fornisce:

- `consentId` id del consenso
- `consentStatus` stato del consenso
- `scaManagerRedirectUrl` URI per effettuare la SCA
- `validUntil` data fine validità

Revoca di un consenso

Consente di rinnovare la SCA (valida tipicamente 90 giorni) a partire da un consenso precedentemente valido.

```
[DELETE] /{itemId}/consents/{consentId}
```

I **parametri** richiesti sono:

- `consentId` path variable - id del consenso.

Nella **response** l'API torna il consenso stesso, con lo status aggiornato.

Recuperare i conti collegati ad un consenso

Consente di recuperare l'elenco dei conti/carte in proprio possesso presso la banca a fronte del consenso precedentemente fornito.

```
[GET] /{itemId}/consents/{consentId}/accounts
```

I **parametri** richiesti sono:

- `consentId` path variable - id del consenso.

Nella **response** l'API fornisce:

- `accounts` lista degli account composti dalle seguenti proprietà
 - `accountId` stato del consenso
 - `resourceId` id della risorsa
 - `iban` codice IBAN
 - `currency` valuta di conto
 - `providerId` id della banca
 - `productCode` codice del prodotto
 - `consentId` id del consenso

Banche e Prodotti

Descrizione

In questa sezione vengono elencati gli endpoint che consentono di ottenere informazioni riguardante le banche e i prodotti associati.

Banche

Consente di recuperare l'elenco delle banche.

```
[GET] /{itemId}/banks
```

Nella **response** l'API fornisce:

- `id` id della banca (o provider) da utilizzare come *bankId* (o *providerId*) nelle varie chiamate
 - `aspspCode` codice ASPSP
 - `satus` stato
 - `businessName` descrizione della banca
 - `aisInfo` informazioni relative al servizio [Conti](#)
 - `globalBankConsent` consenso globale supportato
 - `supportedNatures` nature supportate (conto, carta o entrambi)
 - `logo` URL del logo della banca
 - `readyToUse` indica che la banca è pronta per essere utilizzata dal servizio
 - `supportedPaymentProducts` modalità di pagamento supportate
 - `SCT-IT` SCT Italia
 - `SCT-EU` SCT Europa
 - `IP` SCT istantaneo
 - `CB` Cross-border
 - `T2` Target 2
-

Prodotti

Consente di recuperare l'elenco dei prodotti di una determinata banca.

```
[GET] /{itemId}/banks/{bankId}/products
```

Il parametro `bankId` fa riferimento all'id della banca ottenibile chiamando l'endpoint [Banche](#) sopra.

Nella **response** l'API fornisce:

- `aspspProductCode` codice del prodotto da utilizzare come *productCode* nelle varie chiamate
- `aspspProductDescription` descrizione prodotto estesa
- `aspspProductUuid` uuid del prodotto
- `aspspProductSuggestedLabel` descrizione sintetica del prodotto

Settings

Descrizione

In questa sezione vengono elencati gli endpoint riguardanti le impostazioni/configurazioni di pay-gateway

Configurazione di pagamento

[GET] /{itemId}/settings/config

Response:

- `minChargeAmount` : importo minimo transazione (in centesimi)
- `maxChargeAmount` : importo massimo transazione (in centesimi)
- `balanceRefundReserve` : esposizione massima per rimborso (in centesimi)
- `minWireAmount` : importo minimo accredito (in centesimi)
- `maxWireAmount` : importo massimo accredito (in centesimi)
- `netChargeEnabled` : incasso tramite SDD al netto abilitato
- `payoutBlocked` : accredito bloccato