

Lettura Fatture

Le chiamate per la lettura delle fatture si dividono in due step.

Questo non vuol dire però che sono complementari, ovvero, le chiamate di lettura fatture (step 2) possono essere effettuate anche senza lo step 1.

Perché è importante però utilizzare lo step 1?

Lo step 1, ovvero la chiamata di [ItemOverview](#), è importante perché ci consente di verificare rapidamente se ci sono delle nuove fatture.

Questo permette di non sovraccaricare i nostri server perché se chiamando l'ItemOverview (step 1) ci restituisce che non ci sono delle nuove fatture è inutile procedere con lo step 2.

Dopo aver effettuato la chiamata di ItemOverview possiamo procedere a chiamare le api di lettura per ogni item che ha fatture.

Lista Fatture

[GET/v2/invoices](#)

Parametri da impostare:

- **ownerId:** da popolare con l'identificativo dell'azienda (UUID/CF/P.iva) Se l'azienda ha più di un ufficio e si vogliono recuperare le fatture di tutti gli uffici è possibile aggiungere al valore di ownerId il suffisso **-ALL**.
- **active:** da popolare con *true* o *false*. True se si vuole ottenere la lista delle fatture attive. False se si vuole ottenere la lista delle fatture passive;
- **lastTimestampFrom:** da popolare con la data espressa in millisecondi. Ad eccezione della prima volta che può essere popolato con un data a piacere le volte successive deve essere popolato con il *"lastTimestamp"* della fattura più recente ricevuta dalla response.

Come impostare la chiamata:

Eseguo la prima chiamata con data a piacere, ad esempio 24/01/2022

Dalla response ottengo tutte le fatture che hanno subito aggiornamenti successivi alla data impostata.

Le fatture vengono restituite dalla più recente alla meno recente.

Il giorno seguente ho bisogno di sapere se ci sono altri aggiornamenti sulle fatture?
Quello che bisogna fare è fare è impostare nella chiamata il timestamp dell'ultima fattura ottenuta dalla response precedente.

Esempio:

24/01/2022 ore 11:00 timestamp: 1643022047000

Response:

```
"...  
"currentStatusName": "INVIATO",  
"lastTimestamp": 1643047247000,  
"active": true,  
..."
```

25/01/2022 ore 11:00 timestamp: 1643047247000

In questo modo dalla response otterremo solo le fatture che hanno subito un aggiornamento e non nuovamente tutta la lista delle fatture.

Il comportamento ideale da seguire quindi è sincronizzare le fatture tramite questa chiamata e poi impostare delle query da parte vostra per ottenere specifici risultati.

E' possibile trovare la stessa fattura con *timestamp* diversi?

Sì è possibile perché il *timestamp* di una fattura viene modificato ogniqualvolta essa subisce un aggiornamento.

Continuation Token

Il "continuationToken" ci permette di vedere, qualora siano presenti, le altre fatture ottenute dalla response.

Il "size" della chiamata è impostato di default a 20 in modo da ottenere dalla response non più di 20 record (fatture).

Nel caso in cui siano presenti più di 20 fatture, in fondo alla response, troveremo due tag che ne danno conferma e sono "hasNext" popolato con "true" e "continuationToken" popolato con una stringa.

Esempio:

```
},  
  "page": {  
    "size": 20,  
    "hasNext": true,
```

```
"continuationToken":  
"6g5TtcULxhoHlBd9DlT0mnTlqtnDa5Hz8zlimF7R1ahgTkS2f0YE1z3HM162btsWr8iV0xi2Pbd02nHD7XlwFX3Wtt2cA  
5iTxLJ07-zpwUXV1WJ9HAd9ko3t22M3tjjc"  
}
```

Quindi per visualizzare le altre fatture presenti nella response non dobbiamo far altro che aggiungere alla chiamata il parametro “continuationToken” e popolarlo con la stringa indicata.

Esempio di chiamata:

```
curl --location --request GET 'https://b2bread-api-  
test.agyo.io/api/v2/invoices?ownerId=bf01b67b-86c6-4695-b8c3-  
2135aa0b8d6c&active=true&lastTimestampFrom=1546335945000' \  
--header 'User-Agent: Postman' \  
--header 'X-App-Name: VRxyz' \  
--header 'X-App-Version: 1.0' \  
--header 'Content-Type: application/json' \  
--header 'X-Request-ID: cuid(codice univoco)' \  
--header 'X-Correlation-ID: cuid(codice univoco)' \  
--header 'X-Item-ID: UUID (identificativo azienda)' \  
--header 'X-User-ID: ID Chiave Tecnica' \  
--header 'Accept-Language: it-IT' \  
--header 'Authorization: Bearer '
```

Dettaglio Fattura

[GET/v2/invoices/{hubId}](#)

Esempio chiamata:

```
curl --location --request GET 'https://b2bread-api-test.agyo.io/api/v2/invoices/{hubId}' \  
--header 'accept: application/hal+json' \  
--header 'User-Agent: Postman' \  
--header 'X-App-Name: VRxyz' \  
--header 'X-App-Version: 1.0' \  
--header 'X-Request-ID: cuid (codice univoco)' \  
--header 'X-Correlation-ID: cuid (codice univoco)' \  
--header 'X-Item-ID: Identificativo azienda' \  
--header 'X-User-ID: ID Chiave Tecnica' \  
--header 'Authorization: Bearer '
```

Download Fattura

[GET/v2/invoices/{hubId}/download](https://b2bread-api-test.agyo.io/api/v2/invoices/{hubId}/download)

Permette il download della fattura così come ha raggiunto l'ultimo stato. Questo significa che il download di una fattura inviata a SDI scaricherà una fattura firmata XADES (a meno che l'azienda abbia il flag di invio senza firma) con terzo intermediario TS (a meno che la fattura non sia già stata firmata in upload)

Mentre per la fattura ricevuta da SDI si riceverà una fattura non firmata, quindi in semplice formato xml. Se la fattura fa parte di un lotto si riceve il solo xml della fattura a cui si fa riferimento in request in quanto essa viene incorporata dal lotto per facilità d'uso.

Per i flussi che non passano per SDI si riceve l'xml della fattura relativa all'ultimo stato, quindi se previsto l'embedding degli allegati conterrà anche quelli.

Download del file in PDF: <https://b2bread-api-test.agyo.io/api/v2/invoices/{hubId}/download?format=PDF>

Download del file in ASSOSOFTWARE: <https://b2bread-api-test.agyo.io/api/v2/invoices/{hubId}/download?format=ASSOSW>

Download del file in XML: <https://b2bread-api-test.agyo.io/api/v2/invoices/{hubId}/download?format=XML>

Esempio chiamata:

```
curl --location --request GET 'https://b2bread-api-test.agyo.io/api/v2/invoices/{hubId}/download?format=XML' \
--header 'accept: application/hal+json' \
--header 'User-Agent: Postman' \
--header 'X-App-Name: VRxyz' \
--header 'X-App-Version: 1.0' \
--header 'X-Request-ID: cuid (codice univoco)' \
--header 'X-Correlation-ID: cuid (codice univoco)' \
--header 'X-Item-ID: UUID (identificativo azienda)' \
--header 'X-User-ID: ID Chiave Tecnica' \
--header 'Authorization: Bearer '
```

[GET/v2/invoices/{hubId}/download/original](#)

Permette il download della fattura originale, quindi per una fattura inviata a SDI permettere di scaricare la fattura così come è stata caricata dall'utente.

Mentre per le passive ricevute da SDI permette di scaricare la fattura originale ricevuta, firmata XADES o PADES (p7m).

Per i flussi che non passano per SDI si riceve l'xml caricato dall'utente senza alcuna modifica.

Se la fattura in request fa riferimento ad un lotto si riceverà il lotto intero, in quanto la fattura si trova all'interno di quel file originale.

Download Notifica SDI

[GET/v2/invoices/{hubId}/downloadAllMessages](#)

Esempio Chiamata:

```
curl --location --request GET 'https://b2bread-api-test.agyo.io/api/v2/invoices/{hubId}/downloadAllMessages' \
--header 'accept: application/hal+json' \
--header 'User-Agent: Postman' \
--header 'X-App-Name: VRxyz' \
--header 'X-App-Version: 1.0' \
--header 'X-Request-ID: cuuid' \
--header 'X-Correlation-ID: uuid-v4' \
--header 'X-Item-ID: ID Azienda' \
--header 'X-User-ID: ID Chiave Tecnica' \
--header 'Authorization: Bearer '
```

Notifica di Scarto

[GET/v2/invoices/{hubId}/download](#)

Parametri:

- **messageId**: il valore da inserire in questo parametro è recuperabile dalla response della chiamata di *Dettaglio Fattura*.
Il tag di riferimento è "*notificationId*" e si trova nella sezione "*messages*";
- **format**: scegliere il tipo di formato della notifica (Esempio "XML").

Esempio Chiamata:

```
curl --location --request GET 'https://b2bread-api-  
test.agyo.io/api/v2/invoices/{hubId}/download?messageId=Ivwy1pPrTtV1xA&format=XML' \  
--header 'Authorization: Bearer ' \  
--header 'X-App-Name: VRxyz' \  
--header 'X-App-Version: 1.0' \  
--header 'User-Agent: Postman' \  
--header 'X-Request-ID: cuuid' \  
--header 'X-Correlation-ID: uuid-v4' \  
--header 'X-Item-ID: ID Azienda' \  
--header 'X-User-ID: ID Chiave Tecnica' \  
--header 'Accept-Language: it-IT' \  
--header 'Content-Type: application/json' \  
--header 'Cookie:  
visid_incap_2770748=A9AiR++qR4+3fodrMwxP8h4nWGIAAAAAQUIPAAAAABJFK5guyJfBL1cz/JUufW5;  
visid_incap_2773288=C84tkoe0SrSMtGsu/hERxgKbb2IAAAAAQUIPAAAAAABCra5YhrEW739veNMLkYa '
```

Download Metadata

[GET/v2/invoices/{hubId}/download/metadata](#)

Esempio Chiamata:

```
curl --location --request GET 'https://b2bread-api-  
test.agyo.io/api/v2/invoices/{hubId}/download/metadata' \  
--header 'Authorization: Bearer ' \  
--header 'accept: application/hal+json' \  
--header 'User-Agent: Postman' \  
--header 'X-App-Name: VRxyz' \  
--header 'X-App-Version: 1.0' \  
--header 'X-Request-ID: cuuid' \  
--header 'X-Correlation-ID: uuid-v4' \  
--header 'X-Item-ID: ID Azienda' \  
--header 'X-User-ID: ID Chiave Tecnica' \  

```

Revision #55

Created 2020-05-29 10:13:46 UTC by Zittino

Updated 2024-04-09 13:16:24 UTC by Luca Zilembo