

Shop

Overview

Endpoint per le operazioni sullo shop

Version information

Version : 1.2.0

URI scheme

BasePath : /shop

Tags

- Item e prodotti

Paths

Ricerca degli item secondo alcuni criteri

GET /shop/items/

Description

Ricerca tra gli item quelli che soddisfano i criteri richiesti

Parameters

Type	Name	Description	Schema
Query	<code>page_size</code> <i>required</i>	Numero di item da restituire	number (int)
Query	<code>page_number</code> <i>required</i>	Offset per paginazione	number (int)
Query	<code>category</code> <i>optional</i>	Categoria dell'item	string
Query	<code>name</code> <i>optional</i>	Stringa che l'item deve contenere nel nome	string
Query	<code>price_asc</code> <i>optional</i>	Ordina per prezzo crescente	boolean
Query	<code>price_desc</code> <i>optional</i>	Ordina per prezzo decrescente	boolean
Query	<code>name_asc</code> <i>optional</i>	Ordina per nome in ordine alfabetico	boolean
Query	<code>name_desc</code> <i>optional</i>	Ordina per nome in ordine alfabetico inverso	boolean

Responses

HTTP Code	Description	Schema
200	OK	< Item > array
400	Parametri malformati	ValidationErrorMessage
500	Errore lato server	No Content

Produces

- `application/json`

Tags

- Item e prodotti

Creazione di un nuovo item nello shop

POST /shop/items/

Description

Gestisce l'inserimento di un nuovo item e i prodotti inclusi.

Richiede permessi ["operator", "shop_write"]

Body parameter

Dati dell'item

Name : body

Flags : required

Name	Schema
name <i>required</i>	string
description <i>optional</i>	string
category <i>required</i>	string
relevance <i>optional</i>	number
products <i>required</i>	< products > array

products

Name	Schema
barcode <i>required</i>	string
name <i>required</i>	string
description <i>optional</i>	string
price <i>required</i>	number
quantity <i>required</i>	number
target_species <i>optional</i>	< string > array
images <i>optional</i>	< images > array

images

Name	Schema
<code>path</code> <i>required</i>	string
<code>description</code> <i>required</i>	string

Responses

HTTP Code	Description	Schema
201	Inserito con successo Headers : <code>location</code> (string) : Endpoint da cui è possibile accedere all'item.	Item
400	Parametri malformati	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi insufficienti	ErrorMessage
409	Il barcode di uno dei prodotti è già esistente	FieldErrorMessage
500	Errore lato server	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Item e prodotti

Ricerca di item

```
GET /shop/items/{item_id}
```

Description

Ricerca di un item dato il suo id.

Parameters

Type	Name	Description	Schema
Path	<code>item_id</code> <i>required</i>	Id dell'item	string (ObjectId)

Responses

HTTP Code	Description	Schema
200	Item ricercato	< Item > array
400	Parametri malformati	ValidatorErrorMessage
404	Nessun item trovato	ErrorMessage
500	Errore lato server	No Content

Produces

- `application/json`

Tags

- Item e prodotti

Modifica di un item

```
PUT /shop/items/{item_id}
```

Description

Gestisce la modifica dei dati di un item.

Richiede permessi ["operator", "shop_write"]

Parameters

Type	Name	Description	Schema
Path	item_id <i>required</i>	Id dell'item da modificare	string (ObjectId)

Body parameter

Dati dell'item

Name : body

Flags : optional

Name	Schema
name <i>optional</i>	string
description <i>optional</i>	string
category <i>optional</i>	string
relevance <i>optional</i>	number (int)
products <i>optional</i>	< products > array

products

Name	Schema
barcode <i>optional</i>	string
name <i>optional</i>	string
description <i>optional</i>	string
target_species <i>optional</i>	< string > array
price <i>optional</i>	number
quantity <i>optional</i>	number

Name	Schema
<code>images</code> <i>optional</i>	<code>< images > array</code>

`images`

Name	Schema
<code>path</code> <i>required</i>	<code>string</code>
<code>description</code> <i>required</i>	<code>string</code>

Responses

HTTP Code	Description	Schema
200	Modificato con successo	<code>Item</code>
400	Parametri malformati	<code>ValidationErrorMessage</code>
401	Non autenticato	<code>ErrorMessage</code>
403	Permessi insufficienti	<code>ErrorMessage</code>
404	Nessun item trovato	<code>ErrorMessage</code>
500	Errore lato server	<code>No Content</code>

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Item e prodotti

Cancellazione di un item

```
DELETE /shop/items/{item_id}
```

Description

Gestisce la cancellazione di un item e di tutti i suoi prodotti associati.

Richiede permessi ["operator", "shop_write"]

Parameters

Type	Name	Description	Schema
Path	<code>item_id</code> <i>required</i>	Id dell'item da cancellare	string (ObjectId)

Responses

HTTP Code	Description	Schema
204	Cancellato con successo	No Content
400	Parametri malformati	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi insufficienti	ErrorMessage
404	Nessun item trovato	ErrorMessage
500	Errore lato server	No Content

Tags

- Item e prodotti

Ricerca di un item per barcode di un prodotto

```
GET /shop/items/barcode/{barcode}
```

Description

Ricerca l'item che contiene un determinato prodotto ricercato per barcode.

Richiede permessi ["operator", "shop_read"]

Parameters

Type	Name	Description	Schema
Path	barcode <i>required</i>	Barcode del prodotto	string

Responses

HTTP Code	Description	Schema
200	OK	Item
400	Parametri malformati	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi insufficienti	ErrorMessage
404	Nessun item trovato	ErrorMessage
500	Errore lato server	No Content

Produces

- `application/json`

Tags

- Item e prodotti

Incremento rilevanza

```
POST /shop/items/{item_id}/click
```

Description

Incrementa la rilevanza di un item.

Parameters

Type	Name	Description	Schema
Path	<code>item_id</code> <i>required</i>	Id dell'item a cui incrementare la rilevanza	string (ObjectId)

Responses

HTTP Code	Description	Schema
204	Incrementato con successo	No Content
400	Parametri malformati	ValidationErrorMessage
404	Item inesistente	ErrorMessage
500	Errore interno	No Content

Tags

- Item e prodotti

Lista di tutte le categorie

```
GET /shop/categories/
```

Description

Restituisce la lista di tutte le categorie disponibili.

Responses

HTTP Code	Description	Schema
200	Lista di tutte le categorie disponibili	< Category > array
500	Internal server error	No Content

Produces

- [application/json](#)

Tags

- Categorie

Inserimento di categorie

POST /shop/categories/

Description

Creazione di una nuova categoria.

Richiede permessi ["operator", "shop_write"]

Body parameter

Dati della categoria

Name : body

Flags : required

Type : [Category](#)

Responses

HTTP Code	Description	Schema
201	Creato con successo	Category
400	Input malformato	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi mancanti	ErrorMessage
409	Categoria con lo stesso nome già esistente	FieldErrorMessage
500	Internal server error	No Content

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- Categorie

Aggiornamento di categorie

PUT /shop/categories/{category}

Description

Aggiornamento di una categoria.

Richiede permessi ["operator", "shop_write"]

Parameters

Type	Name	Description	Schema
Path	category <i>required</i>	Categoria da aggiornare	string

Body parameter

Dati da aggiornare della categoria

Name : body

Flags : optional

Type : [Category](#)

Responses

HTTP Code	Description	Schema
200	Aggiornato con successo	Category
400	Input malformato	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi mancanti	ErrorMessage
404	Categoria non esistente	ErrorMessage
500	Internal server error	No Content

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- Categorie

Cancellazione di categorie

```
DELETE /shop/categories/{category}
```

Description

Cancellazione di una categoria.

Richiede permessi ["operator", "shop_write"]

Parameters

Type	Name	Description	Schema
Path	<code>category</code> <i>required</i>	Categoria da cancellare	string

Responses

HTTP Code	Description	Schema
204	Cancellato con successo	No Content
400	Input malformato	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi mancanti	ErrorMessage
404	Categoria non esistente	ErrorMessage
500	Internal server error	No Content

Tags

- Categorie

Ricerca di ordini

```
GET /shop/orders/
```

Description

Gestisce la ricerca degli ordini, ordinati in senso decrescente.

I clienti posso cercare solo i propri ordini, gli operatori tutti.

Richiede permessi [["customer"]], [["admin"], ["operator", "warehouse"]]

Parameters

Type	Name	Description	Schema
Query	<code>page_size</code> <i>required</i>	Numero di post per pagina	number (int)
Query	<code>page_number</code> <i>required</i>	Numero di pagina	number (int)
Query	<code>status</code> <i>optional</i>	Stato dell'ordine	string
Query	<code>start_date</code> <i>optional</i>	Inizio intervallo temporale di ricerca	string (date-time)
Query	<code>end_date</code> <i>optional</i>	Fine intervallo temporale di ricerca	string (date-time)
Query	<code>customer</code> <i>optional</i>	Username dell'autore dell'ordine	string

Responses

HTTP Code	Description	Schema
200	Trovato con successo	< Order > array
400	Parametri malformati	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi insufficienti	ErrorMessage
500	Errore lato server	No Content

Produces

- `application/json`

Tags

- Ordini

Creazione di ordini

POST /shop/orders/

Description

Gestisce la creazione di un nuovo ordine associato al cliente.

Richiede permessi [["customer"]] , [["admin"]]

Body parameter

Name : body

Flags : required

Name	Description	Schema
products <i>required</i>		< products > array
pickup <i>required</i>	true per ritiro in sede, false per consegna	boolean
hub_code <i>optional</i>	Codice dell'hub di ritiro. Obbligatorio se si richiede il ritiro	string
address <i>optional</i>	Indirizzo di consegna. Obbligatorio se si richiede la consegna	Address

products

Name	Schema
barcode <i>required</i>	string
quantity <i>required</i>	number

Responses

HTTP Code	Description	Schema
200	Inserito con successo Headers : location (string) : Endpoint da cui è possibile accedere ai dati dell'ordine.	Order
400	Parametri malformati, quantità insufficiente	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi insufficienti	ErrorMessage
404	Parametri fanno riferimento a entità inesistenti	ErrorMessage
500	Errore lato server	No Content

Consumes

- [application/json](#)

Produces

- `application/json`

Tags

- Ordini

Ricerca ordine per id

```
GET /shop/orders/{order_id}
```

Description

Gestisce la ricerca di uno specifico ordine per id.

I clienti possono ricercare solo i propri ordini, gli operatori tutti.

Richiede permessi [["customer"]], [["admin"], ["operator", "warehouse"]]

Parameters

Type	Name	Description	Schema
Path	<code>order_id</code> <i>required</i>	Id dell'ordine	string (ObjectId)

Responses

HTTP Code	Description	Schema
200	Trovato con successo	Order
400	Parametri malformati	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi insufficienti	ErrorMessage
404	Ordine inesistente	ErrorMessage
500	Errore lato server	No Content

Produces

- `application/json`

Tags

- Ordini

Aggiornamento ordine per id

```
PUT /shop/orders/{order_id}
```

Description

Gestisce l'aggiornamento di uno specifico ordine per id.

Richiede permessi [], [["admin"], ["operator", "warehouse"]]

Parameters

Type	Name	Description	Schema
Path	<code>order_id</code> <i>required</i>	Id dell'ordine	string (ObjectId)

Body parameter

Name : body

Flags : required

Name	Schema
<code>status</code> <i>optional</i>	string

Responses

HTTP Code	Description	Schema
200	Aggiornato con successo	Order
400	Parametri malformati	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi insufficienti	ErrorMessage
404	Ordine inesistente	ErrorMessage
500	Errore lato server	No Content

Produces

- `application/json`

Tags

- Ordini

Cancellazione ordine per id

```
DELETE /shop/orders/{order_id}
```

Description

Gestisce la cancellazione di uno specifico ordine per id.

I clienti possono cancellare i propri ordini che non sono in lavorazione, gli operatori tutti.

Richiede permessi [["customer"]], [["admin"], ["operator", "warehouse"]]

Parameters

Type	Name	Description	Schema
Path	<code>order_id</code> <i>required</i>	Id dell'ordine	string (ObjectId)

Responses

HTTP Code	Description	Schema
204	Cancellato con successo	No Content
400	Parametri malformati	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi insufficienti	ErrorMessage
404	Ordine inesistente	ErrorMessage
500	Errore lato server	No Content

Produces

- `application/json`

Tags

- Ordini

Sessione di pagamento

POST /shop/orders/{order_id}/checkout

Description

Avvia una sessione di pagamento per un ordine.

Richiede permessi [["customer"]], []

Parameters

Type	Name	Description	Schema
Path	<code>order_id</code> <i>required</i>	Id dell'ordine	string (ObjectId)

Responses

HTTP Code	Description	Schema
200	Creato con successo	Response 200
400	Parametri malformati	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi insufficienti	ErrorMessage
404	Ordine inesistente	ErrorMessage
500	Errore lato server	No Content

Response 200

Name	Description	Schema
<code>clientSecret</code> <i>optional</i>	Client secret di Stripe	string

Produces

- `application/json`

Tags

- Ordini

Pagamento ordine

```
POST /shop/orders/{order_id}/success
```

Description

Aggiorna un ordine come pagato.

Nota: NON è un'implementazione sicura!

Parameters

Type	Name	Description	Schema
Path	<code>order_id</code> <i>required</i>	Id dell'ordine	string (ObjectId)

Responses

HTTP Code	Description	Schema
204	Ordine aggiornato con successo	No Content
400	Parametri malformati oppure ordine già pagato	ValidationErrorMessage
404	Ordine inesistente	ErrorMessage
500	Errore lato server	No Content

Tags

- Ordini

Ricerca contenuto carrello

```
GET /users/customers/{username}/cart/
```

Description

Gestisce la ricerca del contenuto nel carrello dell'utente.

Nota: non controlla la presenza di inconsistenze tra quantità richiesta e quantità disponibile.

Richiede permessi [["customer"]], [["admin"]]

Parameters

Type	Name	Schema
Path	<code>username</code> <i>required</i>	string

Responses

HTTP Code	Description	Schema
200	Carrello dell'utente	Cart
400	Parametri malformati	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi insufficienti	ErrorMessage
404	Utente inesistente	ErrorMessage
500	Errore lato server	No Content

Produces

- `application/json`

Tags

- Carrello

Modifica carrello

```
PUT /users/customers/{username}/cart/
```

Description

Gestisce la modifica del contenuto nel carrello

Richiede permessi [["customer"]], [["admin"]]

Parameters

Type	Name	Schema
Path	<code>username</code> <i>required</i>	string

Body parameter

Name : body

Flags : required

Type : < [body](#) > array

body

Name	Schema
<code>barcode</code> <i>required</i>	string
<code>quantity</code> <i>required</i>	number

Responses

HTTP Code	Description	Schema
200	Modificato con successo	Cart
400	Parametri malformati, quantità insufficiente	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi insufficienti	ErrorMessage
404	Utente o prodotto inesistente	ErrorMessage
500	Errore lato server	No Content

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- Carrello

Inserimento in carrello

```
POST /users/customers/{username}/cart/
```

Description

Gestisce l'inserimento di un prodotto nel carrello. Se già presente, la quantità viene aggiornata.

Richiede permessi [["customer"]], [["admin"]]

Parameters

Type	Name	Schema
Path	username <i>required</i>	string

Body parameter

Name : body

Flags : required

Name	Schema
barcode <i>required</i>	string
quantity <i>required</i>	number

Responses

HTTP Code	Description	Schema
200	Inserito con successo	Cart
400	Parametri malformati, quantità insufficiente	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi insufficienti	ErrorMessage
404	Utente o prodotto inesistente	ErrorMessage
500	Errore lato server	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Carrello

Definitions

Category

Name	Description	Schema
name <i>required</i>		string
icon <i>required</i>	Pattern : <code>"^([A-Za-z0-9+/{4})*([A-Za-z0-9+/{2}== [A-Za-z0-9+/{3}=])?\$"</code>	string (byte)

Item

Name	Schema
id <i>required</i>	ObjectId
name <i>required</i>	string
description <i>required</i>	string
category <i>required</i>	string
relevance <i>required</i>	number
products <i>required</i>	< Product > array

Product

Name	Schema
barcode <i>required</i>	string
name <i>required</i>	string
description <i>required</i>	string
images <i>required</i>	< images > array
target_species <i>required</i>	< string > array
price <i>required</i>	number
quantity <i>required</i>	number

images

Name	Schema
path <i>required</i>	string

Name	Schema
description <i>required</i>	string

Order

Name	Description	Schema
id <i>required</i>		ObjectId
customer <i>required</i>	Username del cliente	string
products <i>required</i>	Informazioni sui prodotti dell'ordine (snapshot di ciò che c'era al momento dell'ordine)	< products > array
total <i>required</i>		number
pickup <i>required</i>	true per ritiro in sede, false per consegna	boolean
hub_code <i>optional</i>	Codice dell'hub in cui ritirare	string
address <i>optional</i>	Indirizzo di consegna	string
status <i>required</i>	Stato dell'ordine	string
creationDate <i>required</i>		string (date-time)

products

Name	Schema
barcode <i>required</i>	string
name <i>required</i>	string
price <i>required</i>	number
quantity <i>required</i>	number

Address

Name	Schema
city <i>optional</i>	string
street <i>optional</i>	string
number <i>optional</i>	string
postalcode <i>optional</i>	string

Cart

Type : < [Cart](#) > array

Cart

Name	Schema
<code>source_item</code> <i>optional</i>	Item
<code>product</code> <i>required</i>	Product
<code>quantity</code> <i>required</i>	number

ValidationErrorMessage

Type : < [FieldErrorMessage](#) > array

FieldErrorMessage

Name	Description	Schema
<code>field</code> <i>required</i>	Il campo errato	string
<code>message</code> <i>optional</i>	Informazioni sull'errore	string

ErrorMessage

Name	Description	Schema
<code>message</code> <i>optional</i>	Informazioni sull'errore	string

ObjectId

Type : string (ObjectId)