

# Appuntamenti

## Overview

Endpoint per la gestione di appuntamenti

## Version information

*Version* : 1.0.0

## URI scheme

*BasePath* : /appointments

## Paths

# Ricerca generale

GET /appointments/

## Description

Gestisce la ricerca di tutti gli appuntamenti oppure è possibile applicare un filtro per mostrare gli appuntamenti di un cliente o di un operatore

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

## Parameters

Type	Name	Description	Schema
Query	<code>username</code> <i>optional</i>	Username dell'utente	string

## Responses

HTTP Code	Description	Schema
200	Trovato con successo	< <a href="#">BookingData</a> > array
400	Input malformato	<a href="#">ValidationErrorMe</a> <a href="#">ssage</a>
401	Non autenticato	<a href="#">ErrorMessage</a>
403	Permessi mancanti	<a href="#">ErrorMessage</a>
500	Internal server error	No Content

## Produces

- [application/json](#)

## Tags

- Appointments

# Creazione

POST /appointments/

## Description

Gestisce l'inserimento di un appuntamento

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

## Body parameter

Dati dell'appuntamento

*Name* : body

*Flags* : required

Name	Description	Schema
<b>time_slot</b> <i>required</i>		<a href="#">time_slot</a>
<b>service_id</b> <i>required</i>		<a href="#">ObjectId</a>
<b>customer</b> <i>required</i>	Username del cliente	string
<b>animal_id</b> <i>required</i>		<a href="#">ObjectId</a>
<b>operator</b> <i>required</i>	Username dell'operatore	string
<b>hub</b> <i>required</i>	Codice dell'hub	string

**time\_slot**

Name	Schema
<b>start</b> <i>required</i>	string (date-time)
<b>end</b> <i>required</i>	string (date-time)

## Responses

HTTP Code	Description	Schema
201	Creato con successo Headers : <b>location</b> (string) : Endpoint da cui è possibile accedere ai dati della prenotazione.	<a href="#">BookingData</a>
400	Input malformato	<a href="#">ValidationErrorMessage</a>
401	Non autenticato	<a href="#">ErrorMessage</a>
403	Permessi mancanti	<a href="#">ErrorMessage</a>

HTTP Code	Description	Schema
500	Internal server error	No Content

#### Consumes

- `application/json`

#### Produces

- `application/json`

#### Tags

- Appointments

# Ricerca slot disponibili

GET /appointments/availabilities/

## Description

Restituisce gli slot temporali disponibili per un determinato servizio presso un dato hub.

## Parameters

Type	Name	Description	Schema
Query	<b>start_date</b> <i>required</i>	Data di inizio ricerca	string (date-time)
Query	<b>end_date</b> <i>required</i>	Data di fine ricerca	string (date-time)
Query	<b>hub_code</b> <i>required</i>	Codice dell'hub in cui verificare le disponibilità	string
Query	<b>service_id</b> <i>required</i>	Id del servizio richiesto	string (ObjectId)

## Responses

HTTP Code	Description	Schema
200	Slot disponibili raggruppati per operatore	< <a href="#">TimeSlot</a> > array
400	Input malformato	<a href="#">ValidationErrorMessage</a>
500	Internal server error	No Content

## Produces

- [application/json](#)

## Tags

- Disponibilità

# Ricerca per id

```
GET /appointments/{appointment_id}
```

## Description

Gestisce la ricerca di un appuntamento tramite id

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

## Parameters

Type	Name	Description	Schema
Path	<code>appointment_id</code> <i>required</i>	Id dell'appuntamento	string

## Responses

HTTP Code	Description	Schema
200	Trovato con successo	<a href="#">BookingData</a>
400	Input malformato	<a href="#">ValidationErrorMessage</a>
401	Non autenticato	<a href="#">ErrorMessage</a>
403	Permessi mancanti	<a href="#">ErrorMessage</a>
500	Internal server error	No Content

## Produces

- `application/json`

## Tags

- Appointments

# Cancellazione

```
DELETE /appointments/{appointment_id}
```

## Description

Gestisce la cancellazione di un appuntamento tramite id

Richiede permesso [ ["customer"], ["operator"] ], [ ["admin"] ]

## Parameters

Type	Name	Description	Schema
Path	<code>appointment_id</code> <i>required</i>	Id dell'appuntamento	string

## Responses

HTTP Code	Description	Schema
204	Cancellato con successo	No Content
400	Input malformato	<a href="#">ValidationErrorMessage</a>
401	Non autenticato	<a href="#">ErrorMessage</a>
403	Permessi mancanti	<a href="#">ErrorMessage</a>
404	Non trovato	<a href="#">ErrorMessage</a>
500	Internal server error	No Content

## Produces

- `application/json`

## Tags

- Appointments

# Sessione di pagamento

POST /appointments/{appointment\_id}/checkout

## Description

Avvia una sessione di pagamento per un appuntamento.

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

## Parameters

Type	Name	Schema
Path	appointment_id <i>required</i>	string (ObjectId)

## Responses

HTTP Code	Description	Schema
200	Creato con successo	<a href="#">Response 200</a>
400	Parametri malformati	<a href="#">ValidationErrorMessage</a>
401	Non autenticato	<a href="#">ErrorMessage</a>
403	Permessi insufficienti	<a href="#">ErrorMessage</a>
404	Appuntamento inesistente	<a href="#">ErrorMessage</a>
500	Errore lato server	No Content

### Response 200

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

## Produces

- `application/json`

## Tags

- Appointments



# Pagamento appuntamento

```
POST /appointments/{appointment_id}/success
```

## Description

Aggiorna lo stato di un appuntamento se è stato pagato correttamente.

## Parameters

Type	Name	Schema
Path	<code>appointment_id</code> <i>required</i>	string (ObjectId)

## Responses

HTTP Code	Description	Schema
204	Appuntamento aggiornato con successo	No Content
400	Parametri malformati oppure appuntamento già pagato	<a href="#">ValidationErrorMessage</a>
402	L'appuntamento non è stato pagato	<a href="#">ErrorMessage</a>
404	Appuntamento inesistente	<a href="#">ErrorMessage</a>
500	Errore lato server	No Content

## Tags

- Appointments

# Definitions

## BookingData

Name	Schema
<code>id</code> <i>required</i>	<a href="#">ObjectId</a>
<code>time_slot</code> <i>required</i>	<a href="#">time_slot</a>
<code>service_id</code> <i>required</i>	<a href="#">ObjectId</a>
<code>customer</code> <i>required</i>	string
<code>animal_id</code> <i>required</i>	<a href="#">ObjectId</a>
<code>operator</code> <i>required</i>	string
<code>hub</code> <i>required</i>	string

Name	Schema
<code>paid</code> <i>required</i>	boolean

`time_slot`

Name	Schema
<code>start</code> <i>required</i>	string (date-time)
<code>end</code> <i>required</i>	string (date-time)

## TimeSlot

Name	Description	Schema
<code>time</code> <i>required</i>		<a href="#">time</a>
<code>hub</code> <i>required</i>	Codice dell'hub	string
<code>operator_username</code> <i>required</i>		string

`time`

Name	Schema
<code>start</code> <i>required</i>	string (date-time)
<code>end</code> <i>required</i>	string (date-time)

## 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)