

Post

Overview

Endpoint per le operazioni sui post

Version information

Version : 1.0.0

Paths

Creazione di un nuovo post

POST /posts/

Description

Gestisce l'inserimento di un post. Un utente per pubblicare un post deve avere fatto l'accesso. Richiede permessi ["post_write", "admin"]

Body parameter

Dati del post

Name : body

Flags : optional

Name	Schema
content <i>required</i>	string
topic <i>optional</i>	string
tag_users_id <i>optional</i>	< ObjectId > array
tag_animals_id <i>optional</i>	< ObjectId > array

Responses

HTTP Code	Description	Schema
201	Creato con successo	Post
400	Input malformato	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi mancanti	ErrorMessage
500	Internal server error	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Post

Ricerca dei post secondo determinati criteri

GET /posts/

Description

Gestisce la ricerca dei post dati alcuni criteri. Di default ordina in senso decrescente di data. Gestisce la paginazione.

Parameters

Type	Name	Description	Schema
Query	<code>page_size</code> <i>required</i>	Numero di post per pagina	number
Query	<code>page_number</code> <i>required</i>	Numero di pagina	number
Query	<code>username</code> <i>optional</i>	Username dell'autore	string
Query	<code>topic</code> <i>optional</i>	Categoria del post	string
Query	<code>oldest</code> <i>optional</i>	Visualizzare prima i post più vecchi	boolean

Responses

HTTP Code	Description	Schema
200	Trovato con successo	< Post > array
400	Input malformato	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi mancanti	ErrorMessage
404	Non trovato	ErrorMessage
500	Internal server error	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Post

Inserimento di topic

POST /posts/topics/

Description

Creazione di un nuovo topic.

Richiede permessi [[] , [{"admin"}]]

Body parameter

Dati del topic

Name : body

Flags : required

Type : [Topic](#)

Responses

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

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- [Topic](#)

Lista di tutti i topic

```
GET /posts/topics/
```

Description

Restituisce la lista di tutti i topic disponibili.

Responses

HTTP Code	Description	Schema
200	Ok	< Topic > array
500	Internal server error	No Content

Produces

- [application/json](#)

Tags

- Topic

Aggiornamento di topic

```
PUT /posts/topics/{topic}
```

Description

Aggiornamento di un topic.

Richiede permessi [[] , [{"admin"}]]

Parameters

Type	Name	Description	Schema
Path	topic <i>required</i>	Nome del topic da aggiornare	string

Body parameter

Dati da aggiornare del topic

Name : body

Flags : optional

Name	Schema
name <i>optional</i>	string
icon <i>optional</i>	string (base64)

Responses

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

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Topic

Cancellazione di topic

```
DELETE /posts/topics/{topic}
```

Description

Cancellazione di un topic.

Richiede permessi [[] , [{"admin"}]]

Parameters

Type	Name	Description	Schema
Path	topic <i>required</i>	Nome del topic 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	Topic non esistente	ErrorMessage
500	Internal server error	No Content

Consumes

- `application/json`

Tags

- Topic

Ricerca di un post

```
GET /posts/{post_id}
```

Description

Gestisce la ricerca di un post tramite il suo id. Un utente per ricercare un post deve avere fatto l'accesso. Richiede permessi ["post_write", "admin"]

Body parameter

Id del post

Name : post_id
Flags : required
Type : [ObjectId](#)

Responses

HTTP Code	Description	Schema
200	Trovato con successo	Post
400	Input malformato	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi mancanti	ErrorMessage
404	Non trovato	ErrorMessage
500	Internal server error	No Content

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- Post

Example HTTP request

Request body

```
"61338f5b47bfc65136b5de30"
```

Modifica di un post

```
PUT /posts/{post_id}
```

Description

Gestisce la modifica di un post tramite il suo id. Un utente per modificare un post deve avere fatto l'accesso. Richiede permessi ["post_write", "admin"]

Body parameter

Dati del post

Name : body

Flags : optional

Name	Schema
<code>post_id</code> <i>required</i>	ObjectId
<code>content</code> <i>optional</i>	string
<code>topic</code> <i>optional</i>	string
<code>tag_users_id</code> <i>optional</i>	< ObjectId > array
<code>tag_animals_id</code> <i>optional</i>	< ObjectId > array

Responses

HTTP Code	Description	Schema
200	Modificato con successo	No Content
400	Input malformato	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi mancanti	ErrorMessage
404	Non trovato	ErrorMessage
500	Internal server error	No Content

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- Post

Cancellazione di un post

```
DELETE /posts/{post_id}
```

Description

Gestisce la cancellazione di un post tramite il suo id. Un utente per cancellare un post deve avere fatto l'accesso. Richiede permessi ["post_write", "admin"]

Body parameter

Id del post

Name : post_id
Flags : required
Type : [ObjectId](#)

Responses

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

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- Post

Example HTTP request

Request body

```
"61338f5b47bfc65136b5de30"
```

Creazione di un nuovo commento

```
POST /posts/{post_id}/comments/
```

Description

Gestisce l'inserimento di un commento sotto un post. Un utente per pubblicare un commento deve avere fatto l'accesso. Richiede permessi ["comment_write", "admin"]

Body parameter

Dati del post

Name : body

Flags : optional

Name	Schema
post_id <i>required</i>	ObjectId
content <i>required</i>	string

Responses

HTTP Code	Description	Schema
201	Creato con successo	No Content
400	Input malformato	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi mancanti	ErrorMessage
500	Internal server error	No Content

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- Commento

Ricerca dei commenti

```
GET /posts/{post_id}/comments/
```

Description

Gestisce la ricerca dei commenti di un post tramite il suo id. Un utente per ricercare dei commenti deve avere fatto l'accesso. Richiede permessi ["comment_write", "admin"]

Body parameter

Id del post

Name : post_id
Flags : required
Type : [ObjectId](#)

Responses

HTTP Code	Description	Schema
200	Trovato con successo	< Response 200 > array
400	Input malformato	ValidationErrorMe ssage
401	Non autenticato	ErrorMessage
403	Permessi mancanti	ErrorMessage
404	Non trovato	ErrorMessage
500	Internal server error	No Content

Response 200

Name	Schema
user_id <i>required</i>	ObjectId
content <i>required</i>	string
creationDate <i>required</i>	string (date-time)
updateDate <i>optional</i>	string (date-time)

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- Commento

Example HTTP request

Request body

```
"61338f5b47bfc65136b5de30"
```

Ricerca di un commento

```
GET /posts/{post_id}/comments/{comment_index}
```

Description

Gestisce la ricerca di un commento di un post tramite l'id del post e l'indice del commento nell'array. Un utente per ricercare dei commenti deve avere fatto l'accesso. Richiede permessi ["comment_write", "admin"]

Body parameter

Dati del commento

Name : body

Flags : optional

Name	Schema
<code>post_id</code> <i>required</i>	ObjectId
<code>comment_index</code> <i>required</i>	number (int)

Responses

HTTP Code	Description	Schema
200	Trovato con successo	Response 200
400	Input malformato	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi mancanti	ErrorMessage
404	Non trovato	ErrorMessage
500	Internal server error	No Content

Response 200

Name	Schema
<code>user_id</code> <i>required</i>	ObjectId
<code>content</code> <i>required</i>	string
<code>creationDate</code> <i>required</i>	string (date-time)
<code>updateDate</code> <i>optional</i>	string (date-time)

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Commento

Modifica di un commento

```
PUT /posts/{post_id}/comments/{comment_index}
```

Description

Gestisce la modifica di un commento di un post tramite l'id del post e l'indice del commento nell'array. Un utente per modificare un commento deve avere fatto l'accesso. Richiede permessi ["comment_write", "admin"]

Body parameter

Dati del commento

Name : body

Flags : optional

Name	Schema
<code>post_id</code> <i>required</i>	ObjectId
<code>comment_index</code> <i>required</i>	number (int)
<code>content</code> <i>optional</i>	string

Responses

HTTP Code	Description	Schema
200	Modificato con successo	No Content
400	Input malformato	ValidationErrorMessage
401	Non autenticato	ErrorMessage
403	Permessi mancanti	ErrorMessage
404	Non trovato	ErrorMessage
500	Internal server error	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Commento

Cancellazione di un commento

```
DELETE /posts/{post_id}/comments/{comment_index}
```

Description

Gestisce la cancellazione di un commento di un post tramite l'id del post e l'indice del commento nell'array. Un utente per cancellare un commento deve avere fatto l'accesso. Richiede permessi ["comment_write", "admin"]

Body parameter

Dati del commento

Name : body

Flags : optional

Name	Schema
post_id <i>required</i>	ObjectId
comment_index <i>required</i>	number (int)

Responses

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

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- Commento

Definitions

Topic

Name	Schema
<code>name</code> <i>required</i>	string
<code>icon</code> <i>optional</i>	string (base64)

Post

Name	Schema
<code>user_id</code> <i>required</i>	ObjectId
<code>content</code> <i>required</i>	string
<code>comments</code> <i>optional</i>	< comments > array
<code>topic</code> <i>optional</i>	string
<code>tag_users_id</code> <i>optional</i>	< ObjectId > array
<code>tag_animals_id</code> <i>optional</i>	< ObjectId > array
<code>creationDate</code> <i>required</i>	string (date-time)

comments

Name	Schema
<code>user_id</code> <i>required</i>	ObjectId
<code>content</code> <i>required</i>	string
<code>creationDate</code> <i>required</i>	string (date-time)
<code>updateDate</code> <i>optional</i>	string (date-time)

ObjectId

Type : string ([ObjectId](#))

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