

Post

Overview

Endpoint per le operazioni sui post

Version information

Version : 1.0.0

Paths

Ricerca dei post secondo determinati criteri

```
GET /blog/posts/
```

Description

Gestisce la ricerca dei post dati alcuni criteri. Gestisce la paginazione.

Di default ordina in senso decrescente di data.

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>authors</code> <i>optional</i> | Username degli autori | < string > array |
| Query | <code>topic</code> <i>optional</i> | Topic 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 |
| 500 | Internal server error | No Content |

Produces

- `application/json`

Tags

- Post

Creazione di un nuovo post

POST /blog/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 : required

| Name | Schema |
|-----------------------------------|------------------------------------|
| content <i>required</i> | string |
| topic <i>required</i> | string |
| tag_users <i>optional</i> | < string > array |
| tag_animals_id <i>optional</i> | < ObjectId > array |
| images <i>optional</i> | < images > array |

images

| Name | Schema |
|--------------------------------|--------|
| path <i>required</i> | string |
| description <i>optional</i> | string |

Responses

| HTTP Code | Description | Schema |
|-----------|---|--|
| 201 | Creato con successo Headers : location (string) : Endpoint da cui è possibile accedere al post. | Post |
| 400 | Input malformato | ValidationErrorMessage |
| 401 | Non autenticato | ErrorMessage |
| 403 | Permessi mancanti | ErrorMessage |
| 404 | Topic inesistente | ErrorMessage |

| HTTP Code | Description | Schema |
|-----------|-----------------------|------------|
| 500 | Internal server error | No Content |

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Post

Ricerca di un post

```
GET /blog/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"]]

Parameters

| Type | Name | Description | Schema |
|------|---|-------------|-------------------|
| Path | <code>post_id</code> <i>required</i> | Id del post | string (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 |

Produces

- `application/json`

Tags

- Post

Modifica di un post

```
PUT /blog/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"]]

Parameters

| Type | Name | Description | Schema |
|------|---|---------------------------|-------------------|
| Path | <code>post_id</code> <i>required</i> | Id del post da modificare | string (ObjectId) |

Body parameter

Dati del post

Name : body

Flags : required

| Name | Description | Schema |
|--|---|------------------------------------|
| <code>content</code> <i>optional</i> | | string |
| <code>topic</code> <i>optional</i> | | string |
| <code>tag_users</code> <i>optional</i> | | < string > array |
| <code>tag_animals_id</code> <i>optional</i> | | < ObjectId > array |
| <code>images</code> <i>optional</i> | Immagini aggiornate. Quelle non più presenti verranno cancellate. | < images > array |

images

| Name | Schema |
|---|--------|
| <code>path</code> <i>required</i> | string |
| <code>description</code> <i>optional</i> | string |

Responses

| HTTP Code | Description | Schema |
|-----------|-------------------------|--|
| 200 | Modificato con successo | Post |
| 400 | Input malformato | ValidationErrorMessage |

| HTTP Code | Description | Schema |
|-----------|--------------------------|------------------------------|
| 401 | Non autenticato | ErrorMessage |
| 403 | Permessi mancanti | ErrorMessage |
| 404 | Topic o post non trovato | ErrorMessage |
| 500 | Internal server error | No Content |

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Post

Cancellazione di un post

```
DELETE /blog/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"]]

Parameters

| Type | Name | Description | Schema |
|------|---|---------------------------|-------------------|
| Path | <code>post_id</code> <i>required</i> | Id del post da cancellare | string (ObjectId) |

Responses

| HTTP Code | Description | Schema |
|-----------|-------------------------|--|
| 204 | 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 |

Produces

- `application/json`

Tags

- Post

Caricamento di immagini per post

POST /blog/posts/images/

Description

Gestisce il caricamento di immagini da utilizzare nei post.

Richiede permessi [["post_write"], ["admin"]]

Parameters

| Type | Name | Schema |
|----------|--------------------------|--------|
| FormData | files <i>required</i> | file |

Responses

| HTTP Code | Description | Schema |
|-----------|------------------------------|--|
| 200 | Nome delle immagini inserite | < string > array |
| 400 | Parametri malformati | ValidationErrorMessage |
| 401 | Non autenticato | ErrorMessage |
| 403 | Permessi insufficienti | ErrorMessage |
| 500 | Errore lato server | No Content |

Consumes

- `multipart/form-data`

Produces

- `application/json`

Tags

- Post

Ricerca dei commenti

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

Description

Gestisce la ricerca con paginazione dei commenti di un post tramite il suo id.

Parameters

| Type | Name | Description | Schema |
|-------|---|-------------------------------|-------------------|
| Path | <code>post_id</code> <i>required</i> | Id del post | string (ObjectId) |
| Query | <code>page_size</code> <i>required</i> | Numero di commenti per pagina | number (int) |
| Query | <code>page_number</code> <i>required</i> | Numero di pagina | number (int) |

Responses

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

Produces

- `application/json`

Tags

- Commento

Creazione di un nuovo commento

```
POST /blog/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"]]

Parameters

| Type | Name | Description | Schema |
|------|---|-------------|-------------------|
| Path | <code>post_id</code> <i>required</i> | Id del post | string (ObjectId) |

Body parameter

Dati del commento

Name : body

Flags : required

| Name | Schema |
|---|--------|
| <code>content</code> <i>required</i> | string |

Responses

| HTTP Code | Description | Schema |
|-----------|--|--|
| 201 | Creato con successo Headers : <code>location</code> (string) : Endpoint da cui è possibile accedere al commento. | Comment |
| 400 | Input malformato | ValidationErrorMessage |
| 401 | Non autenticato | ErrorMessage |
| 403 | Permessi mancanti | ErrorMessage |
| 404 | Post inesistente | ErrorMessage |
| 500 | Internal server error | No Content |

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Commento

Ricerca di un commento

```
GET /blog/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.

Parameters

| Type | Name | Description | Schema |
|------|---|--|-------------------|
| Path | <code>post_id</code> <i>required</i> | Id del post | string (ObjectId) |
| Path | <code>comment_index</code> <i>required</i> | Indice del commento all'interno del post | number (int) |

Responses

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

Produces

- `application/json`

Tags

- Commento

Modifica di un commento

```
PUT /blog/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"]]

Parameters

| Type | Name | Description | Schema |
|------|---|--|-------------------|
| Path | <code>post_id</code> <i>required</i> | Id del post | string (ObjectId) |
| Path | <code>comment_index</code> <i>required</i> | Indice del commento all'interno del post | number (int) |

Body parameter

Dati del commento

Name : body

Flags : required

| Name | Schema |
|---|--------|
| <code>content</code> <i>optional</i> | string |

Responses

| HTTP Code | Description | Schema |
|-----------|-------------------------|--|
| 200 | Modificato con successo | Comment |
| 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 /blog/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"]

Parameters

| Type | Name | Description | Schema |
|------|---|--|-------------------|
| Path | <code>post_id</code> <i>required</i> | Id del post | string (ObjectId) |
| Path | <code>comment_index</code> <i>required</i> | Indice del commento all'interno del post | number (int) |

Responses

| HTTP Code | Description | Schema |
|-----------|-------------------------|--|
| 204 | 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 |

Tags

- Commento

Lista di tutti i topic

```
GET /blog/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

Inserimento di topic

POST /blog/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](#)

Aggiornamento di topic

```
PUT /blog/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 : required

| 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 /blog/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

Definitions

Topic

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

Post

| Name | Description | Schema |
|--|----------------------|------------------------------------|
| id <i>required</i> | | ObjectId |
| author <i>required</i> | Username dell'autore | string |
| content <i>required</i> | | string |
| topic <i>required</i> | | string |
| tag_users <i>required</i> | | < string > array |
| tag_animals_id <i>required</i> | | < ObjectId > array |
| images <i>optional</i> | | < images > array |
| creationDate <i>required</i> | | string (date-time) |

images

| Name | Schema |
|---------------------------------------|--------|
| path <i>required</i> | string |
| description <i>optional</i> | string |

Comment

| Name | Description | Schema |
|--|---|--------------------|
| index <i>required</i> | Indice del commento rispetto al post a cui appartiene | number (int) |
| author <i>required</i> | | string |
| content <i>required</i> | | string |
| creationDate <i>required</i> | | string (date-time) |
| updateDate <i>required</i> | | string (date-time) |

ValidationErrorMessage

Type : < [FieldErrorMessage](#) > array

FieldErrorMessage

| Name | Description | Schema |
|---------------------------------|-----------------|--------|
| field <i>required</i> | Il campo errato | string |

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

ErrorMessage

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

ObjectId

Type : string (ObjectId)