

# User

## Overview

Endpoint per le operazioni sugli user

## Version information

*Version* : 1.0.0

## Paths

# Creazione di un nuovo cliente

POST /users/customers/

## Description

Gestisce l'inserimento di un cliente

## Body parameter

Dati del cliente

*Name* : body

*Flags* : required

Name	Schema
username <i>required</i>	string
password <i>required</i>	string
email <i>required</i>	string (email)
name <i>required</i>	string
surname <i>required</i>	string
picture <i>optional</i>	string
gender <i>optional</i>	enum (m, f, altro)
phone <i>optional</i>	string
address <i>optional</i>	<a href="#">Address</a>

## Responses

HTTP Code	Description	Schema
201	Creato con successo Headers : <b>location</b> (string) : Endpoint da cui è possibile accedere ai dati dell'utente.	<a href="#">CustomerUserAll</a>
400	Input malformato	<a href="#">ValidationErrorMessage</a>
409	Cliente con stesso username/email esistente	<a href="#">FieldErrorMessage</a>
500	Internal server error	No Content

## Consumes

- **application/json**

## Produces

- `application/json`

## Tags

- Clienti

# Ricerca di un cliente

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

## Description

Gestisce la ricerca di tutti i dati di un cliente tramite username

Richiede permessi [ ["user"], ["admin"] ]

## Parameters

Type	Name	Description	Schema
Query	<code>username</code> <i>required</i>	Username del cliente	string

## Responses

HTTP Code	Description	Schema
200	Trovato con successo	<a href="#">CustomerUserAll</a>
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

- Clienti

# Modifica di un cliente

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

## Description

Gestisce la modifica di un cliente tramite criterio di ricerca con username

Richiede permessi [ ["user"], ["admin"] ]

## Parameters

Type	Name	Description	Schema
Query	<b>username</b> <i>required</i>	Username del cliente	string

## Body parameter

Dati del cliente

*Name* : body

*Flags* : required

Name	Description	Schema
<b>password</b> <i>optional</i>		string
<b>email</b> <i>optional</i>		string
<b>name</b> <i>optional</i>		string
<b>surname</b> <i>optional</i>		string
<b>picture</b> <i>optional</i>		string
<b>gender</b> <i>optional</i>		string
<b>phone</b> <i>optional</i>		string
<b>address</b> <i>optional</i>		<a href="#">Address</a>
<b>enabled</b> <i>optional</i>	Account abilitato (modificabile solo dagli amministratori)	boolean
<b>permissions</b> <i>optional</i>	Permessi dell'utente (modificabile solo dagli amministratori)	< string > array

## Responses

HTTP Code	Description	Schema
200	Modificato con successo	<a href="#">CustomerUserAll</a>

HTTP Code	Description	Schema
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

### Consumes

- `application/json`

### Produces

- `application/json`

### Tags

- Clienti

# Cancellazione di un cliente

```
DELETE /users/customers/{username}
```

## Description

Gestisce la cancellazione di un cliente tramite criterio di ricerca con username

Richiede permessi [ ["user"], ["admin"] ]

## Parameters

Type	Name	Description	Schema
Query	<code>username</code> <i>required</i>	Username del cliente	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

- Clienti

# Creazione di un nuovo operatore

POST /users/operators/

## Description

Gestisce l'inserimento di un operatore

Richiede permesso [ ["admin"] ]

## Body parameter

Dati dell'operatore

*Name* : body

*Flags* : required

Name	Schema
<b>username</b> <i>required</i>	string
<b>password</b> <i>required</i>	string
<b>email</b> <i>required</i>	string (email)
<b>name</b> <i>required</i>	string
<b>surname</b> <i>required</i>	string
<b>picture</b> <i>optional</i>	string
<b>gender</b> <i>optional</i>	enum (m, f, altro)
<b>phone</b> <i>optional</i>	string
<b>permissions</b> <i>optional</i>	< string > array
<b>role</b> <i>optional</i>	string
<b>services_id</b> <i>optional</i>	< <a href="#">ObjectId</a> > array
<b>working_time</b> <i>required</i>	<a href="#">WorkingTime</a>

## Responses

HTTP Code	Description	Schema
201	Creato con successo Headers : <b>location</b> (string) : Endpoint da cui è possibile accedere ai dati dell'utente.	<a href="#">OperatorUserAll</a>



HTTP Code	Description	Schema
400	Input malformato	<a href="#">ValidationErrorMessage</a>
401	Non autenticato	<a href="#">ErrorMessage</a>
403	Permessi mancanti	<a href="#">ErrorMessage</a>
409	Operatore esistente	<a href="#">FieldErrorMessage</a>
500	Internal server error	No Content

### Consumes

- `application/json`

### Produces

- `application/json`

### Tags

- Operatori

# Ricerca di un operatore

```
GET /users/operators/{username}
```

## Description

Gestisce la ricerca di un operatore tramite username

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

## Parameters

Type	Name	Description	Schema
Query	<code>username</code> <i>required</i>	Username dell'operatore	string

## Responses

HTTP Code	Description	Schema
200	Trovato con successo	<a href="#">OperatorUserAll</a>
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

- Operatori

# Modifica di un operatore

```
PUT /users/operators/{username}
```

## Description

Gestisce la modifica di un operatore tramite criterio di ricerca con username

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

## Parameters

Type	Name	Description	Schema
Query	<b>username</b> <i>required</i>	Username dell'operatore	string

## Body parameter

Dati dell'operatore

*Name* : body

*Flags* : required

Name	Description	Schema
<b>password</b> <i>optional</i>		string
<b>email</b> <i>optional</i>		string
<b>name</b> <i>optional</i>		string
<b>surname</b> <i>optional</i>		string
<b>picture</b> <i>optional</i>		string
<b>gender</b> <i>optional</i>		string
<b>phone</b> <i>optional</i>		string
<b>role</b> <i>optional</i>		string
<b>services_id</b> <i>optional</i>		< <a href="#">ObjectId</a> > array
<b>enabled</b> <i>optional</i>		boolean
<b>permissions</b> <i>optional</i>	Permessi dell'utente (modificabile solo dagli amministratori)	< string > array
<b>working_time</b> <i>optional</i>		<a href="#">WorkingTime</a>

## Responses

HTTP Code	Description	Schema
200	Modificato con successo	<a href="#">OperatorUserAll</a>
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

## Consumes

- `application/json`

## Produces

- `application/json`

## Tags

- Operatori

# Cancellazione di un operatore

```
DELETE /users/operators/{username}
```

## Description

Gestisce la cancellazione di un operatore tramite criterio di ricerca con username

Richiede permessi [ ["admin"] ]

## Parameters

Type	Name	Description	Schema
Query	<code>username</code> <i>required</i>	Username dell'operatore	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

- Operatori

# Ricerca del profilo pubblico di un utente

```
GET /users/profiles/{username}
```

## Description

Gestisce la ricerca dei dati pubblici di un utente

## Parameters

Type	Name	Description	Schema
Query	<b>username</b> <i>required</i>	Username dell'utente	string

## Responses

HTTP Code	Description	Schema
200	Trovato con successo	<a href="#">Response 200</a>
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

### Response 200

Name	Schema
<b>ifCustomer</b> <i>optional</i>	<a href="#">CustomerUserProfile</a>
<b>ifOperator</b> <i>optional</i>	<a href="#">OperatorUserProfile</a>

## Produces

- `application/json`

## Tags

- Profilo pubblico

# Ricerca dei dati di tutti i permessi

```
GET /users/permissions/
```

## Description

Restituisce i dati associati a tutti i permessi.

Richiede autenticazione

## Responses

HTTP Code	Description	Schema
200	Trovato con successo	< <a href="#">Response 200</a> > array
401	Non autenticato	<a href="#">ErrorMessage</a>
500	Internal server error	No Content

### Response 200

Name	Schema
name <i>required</i>	string
url <i>optional</i>	string

## Produces

- [application/json](#)

## Tags

- Permessi

# Ricerca dei dati di un permesso

```
GET /users/permissions/{permission_name}
```

## Description

Restituisce i dati associati ad un permesso.

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

## Parameters

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

## Responses

HTTP Code	Description	Schema
200	Trovato con successo	<a href="#">Response 200</a>
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

### Response 200

Name	Schema
<code>name</code> <i>required</i>	string
<code>url</code> <i>optional</i>	string

## Produces

- `application/json`

## Tags

- Permessi



# Attivazione utenza cliente

PUT /users/customers/enable-me

## Description

Gestisce l'attivazione dell'utenza di un cliente.

Richiede permessi [ ["to\_activate\_user"] ], []

## Responses

HTTP Code	Description	Schema
200	Attivato con successo	<a href="#">CustomerUserAll</a>
401	Non autenticato	<a href="#">ErrorMessage</a>
403	Utenza già attiva	<a href="#">ErrorMessage</a>
404	Utenza inesistente	<a href="#">ErrorMessage</a>
500	Internal server error	No Content

## Produces

- `application/json`

## Tags

- Clienti

## Invio email verifica account

```
GET /users/customers/{username}/verification-mail
```

### Description

Gestisce il (ri)invio della mail di verifica dell'account del cliente.

### Responses

HTTP Code	Description	Schema
204	Inviato con successo	No Content
403	Utenza già attiva	<a href="#">ErrorMessage</a>
404	Utenza inesistente	<a href="#">ErrorMessage</a>
500	Internal server error	No Content

### Tags

- Clienti

# Disponibilità username

```
GET /users/usernames/available/{username}
```

## Description

Verifica se un dato username è disponibile

## Parameters

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

## Responses

HTTP Code	Description	Schema
200	Ok	<a href="#">Response 200</a>
400	Input malformato	<a href="#">ValidationErrorMessage</a>
500	Internal server error	No Content

### Response 200

Name	Description	Schema
<code>available</code> <i>required</i>	true se disponibile, false altrimenti	boolean

## Tags

- Utenti

# Disponibilità email

```
GET /users/emails/available/{email}
```

## Description

Verifica se una dato email è disponibile

## Parameters

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

## Responses

HTTP Code	Description	Schema
200	Ok	<a href="#">Response 200</a>
400	Input malformato	<a href="#">ValidationErrorMessage</a>
500	Internal server error	No Content

### Response 200

Name	Description	Schema
available <i>required</i>	true se disponibile, false altrimenti	boolean

## Tags

- Utenti

# Definitions

## CustomerUserAll

Name	Schema
username <i>required</i>	string
email <i>required</i>	string
name <i>required</i>	string
surname <i>required</i>	string
picture <i>optional</i>	string

Name	Schema
<b>gender</b> <i>optional</i>	string
<b>phone</b> <i>optional</i>	string
<b>permissions</b> <i>required</i>	< string > array
<b>enabled</b> <i>required</i>	boolean
<b>address</b> <i>optional</i>	<a href="#">Address</a>

## CustomerUserProfile

Name	Schema
<b>username</b> <i>required</i>	string
<b>name</b> <i>required</i>	string
<b>surname</b> <i>required</i>	string
<b>picture</b> <i>required</i>	string

## OperatorUserAll

Name	Schema
<b>username</b> <i>required</i>	string
<b>email</b> <i>required</i>	string
<b>name</b> <i>required</i>	string
<b>surname</b> <i>required</i>	string
<b>picture</b> <i>optional</i>	string
<b>gender</b> <i>optional</i>	string
<b>phone</b> <i>required</i>	string
<b>permissions</b> <i>required</i>	< string > array
<b>enabled</b> <i>required</i>	boolean
<b>role</b> <i>required</i>	string
<b>services_id</b> <i>required</i>	< <a href="#">ObjectId</a> > array

Name	Schema
<b>working_time</b> <i>required</i>	<a href="#">WorkingTime</a>

## OperatorUserProfile

Name	Schema
<b>username</b> <i>required</i>	string
<b>name</b> <i>required</i>	string
<b>surname</b> <i>required</i>	string
<b>picture</b> <i>optional</i>	string
<b>email</b> <i>required</i>	string
<b>phone</b> <i>required</i>	string
<b>role</b> <i>required</i>	string
<b>services_id</b> <i>required</i>	< <a href="#">ObjectId</a> > array

## Address

Name	Schema
<b>city</b> <i>optional</i>	string
<b>street</b> <i>optional</i>	string
<b>number</b> <i>optional</i>	string
<b>postal_code</b> <i>optional</i>	string

## WorkingSlot

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

time

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

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

## WorkingTime

Name	Schema
<b>monday</b> <i>required</i>	< <a href="#">WorkingSlot</a> > array
<b>tuesday</b> <i>required</i>	< <a href="#">WorkingSlot</a> > array
<b>wednesday</b> <i>required</i>	< <a href="#">WorkingSlot</a> > array
<b>thursday</b> <i>required</i>	< <a href="#">WorkingSlot</a> > array
<b>friday</b> <i>required</i>	< <a href="#">WorkingSlot</a> > array
<b>saturday</b> <i>required</i>	< <a href="#">WorkingSlot</a> > array
<b>sunday</b> <i>required</i>	< <a href="#">WorkingSlot</a> > array

## ValidationErrorMessage

Type : < [FieldErrorMessage](#) > array

## FieldErrorMessage

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

## ErrorMessage

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