- Web Interfaces API -

Platform for selling and buying used products.

Tabla de contenido

[**Methods for users** 3](#_Toc33107245)

[**Methods for items** 4](#_Toc33107246)

[Methods only available for logged users 4](#_Toc33107247)

[Methods for non-logged users 4](#_Toc33107248)

## **Methods for users**

#### POST /users

Register a new user into the system.

*Request description*

|  |  |  |
| --- | --- | --- |
| userName | string | Publicly visible username |
| name | string | First and family name |
| password | string | Password created by the user. It will have a Bcrypt encryptation. |
| city | string | User home city |
| country | string | User home country |
| email | string | Valid email address for the user |
| isAdmin | boolean | Indicates if the user is admin or not. |
| postings | array | Contains the id’s of all the postings this user has created. By default is null |

*Request body example*

{

"userName": "Nacchofer",

"name": "Ignacio Ferrer",

"password": "pepitoElPapas",

"city": "Valencia",

"country": "Spain",

"email": [fercho@demo.com](mailto:fercho@demo.com)

"isAdmin": false

"postings": []

}

*Response*

|  |  |
| --- | --- |
| 200 | User created successfully |
| 400 | Missing or incorrect data in the request |

*Response data structure description*

|  |  |  |
| --- | --- | --- |
| id\_ | number | The id assigned to the created user |

*Response body example*

{

"id\_ ": 46783265

}

#### GET /users/:id\_user

## **Methods for items**

### Methods only available for logged users

#### POST /users/:id\_user/items

Register a new posting in the system

|  |  |  |
| --- | --- | --- |
| title | string | Name of the item to be sold |
| description | string | Short description of the item |
| category | string | Name of the category in which this item can be found. |
| ***location(\*)*** | object | This object contains different values of the location of posting such as city, country, etc.. |
| images | array | Contains the images of the item (Max. 4 pictures) |
| askingPrice | number | Price for the posting |
| currency | string | Type of currency. It can be “euro” or “dollar”. |
| dateOfPosting | string | Date of posting of the item, in format (DD/MM/YYYY) |
| deliveryType | string | Type of delivery in case that the item is sold. It can be *Shipping* or  *Pickup.* |
| contactInfo | object | Information for contacting the seller. It contains phone number and e-mail |

In the case of the object ***location(\*)***, it contains the following values:

|  |  |  |
| --- | --- | --- |
| city | string | Name of the city where the posting is. |
| country | string | Name of the country of the posting |
| address | string | Address of the posting. |

In the case of the array “*images*” it contains a list of maximum 4 objects, (image) which has these values:

*To be continued….*

In the case of the object “*contactInfo”* it contains the following values:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| sellerName | string | | Name of the seller and family name. | |
| phoneNumber | | number | | Number of the seller containing 9 digits |
| sellerEmail | | string | | E-mail of the seller |

*Request body example*

{

"title": "Teddy bear",

"description": "Teddy bear",

"category": "Rasberry PI",

"location": {

"city": "Valencia"

"country": "Spain"

"address": "Calle de la Amargura"

},

"askingPrice": 36,

"currency": "euro",

"dateOfPosting": "25/11/2019",

"deliveryType": "Shipping",

"contactInfo": {

"sellerName": "Ignacio Ferrer Sanz"

"phoneNumber": 123456789

"sellerEmail": "ignacio@gmail.com"

},

}

*Response*

Status codes

|  |  |
| --- | --- |
| 201 | Senso created successfully |
| 400 | Missing or incorrect data in the request |

*Response data structure description*

|  |  |  |
| --- | --- | --- |
| id | number | The id assigned to the created posting |

*Response body example*

{

"id": 1111111

}

This id will be automatically stored in the array ***postings***, of the user who created it.

#### GET /api/users/:id\_user/items/:id\_item

#### PUT /api/users/:id\_user/items/:id\_item

#### DELETE /api/users/:id\_user/items/:id\_item

A posting will be deleted from the database, and the reference to this posting (found in the posting array of each user) will also be deleted.

### Methods for non-logged users

#### GET /search\_items/:category

#### GET /search\_items/:location

#### GET /search\_items/:dateOfPosting