

Api Documentation

Overview

Api Documentation

Version information

Version : 1.0

License information

License : Apache 2.0

License URL : <http://www.apache.org/licenses/LICENSE-2.0>

Terms of service : urn:tos

URI scheme

Host : localhost:8083

BasePath : /

Tags

- category-controller : Category Controller
- order-details-controller : Order Details Controller
- product-controller : Product Controller
- supplier-controller : Supplier Controller
- user-controller : User Controller
- warehouse-controller : Warehouse Controller

Paths

Creates a new Category.

POST /categories

Parameters

Type	Name	Description	Schema
Body	category <i>required</i>	category	Category

Responses

HTTP Code	Description	Schema
200	OK	Category
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- [application/json](#)

Produces

- [*/*](#)

Tags

- category-controller

Returns list of all Categories in the system.

GET /categories

Responses

HTTP Code	Description	Schema
200	OK	< Category > array
401	Unauthorized	No Content

HTTP Code	Description	Schema
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- category-controller

Returns a specific Category by their identifier. 404 if does not exist.

```
GET /categories/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	Category
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- category-controller

Updates Category.

PUT /categories/{id}

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)
Body	category <i>required</i>	category	Category

Responses

HTTP Code	Description	Schema
200	OK	Category
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- [application/json](#)

Produces

- [*/*](#)

Tags

- category-controller

Deletes a Category from the system. 404 if the category's identifier is not found.

```
DELETE /categories/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
204	No Content	No Content
401	Unauthorized	No Content
403	Forbidden	No Content

Produces

- `application/json`

Tags

- category-controller

Returns Category of a specific Product by their identifier. 404 if does not exist.

```
GET /category/product/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- **/**

Tags

- product-controller

Deletes a User from the system. 404 if the user's identifier is not found.

```
DELETE /deleteUser/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity

HTTP Code	Description	Schema
204	No Content	No Content
401	Unauthorized	No Content
403	Forbidden	No Content

Produces

- `application/json`

Tags

- user-controller

Deletes a Warehouse from the system. 404 if the warehouse's identifier is not found.

```
DELETE /deleteWarehouse/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
204	No Content	No Content
401	Unauthorized	No Content
403	Forbidden	No Content

Produces

- `application/json`

Tags

- `warehouse-controller`

Creates a new Order detail.

POST /order-details

Parameters

Type	Name	Description	Schema
Body	order <i>required</i>	order	OrderDetail

Responses

HTTP Code	Description	Schema
200	OK	OrderDetail
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `*/*`

Tags

- `order-details-controller`

Returns list of all Order details in the system.

```
GET /order-details
```

Responses

HTTP Code	Description	Schema
200	OK	< OrderDetail > array
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- order-details-controller

Returns a specific Order detail by their identifier. 404 if does not exist.

```
GET /order-details/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	OrderDetail
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- **/**

Tags

- order-details-controller

Updates Order detail.

```
PUT /order-details/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)
Body	order <i>required</i>	order	OrderDetail

Responses

HTTP Code	Description	Schema
200	OK	OrderDetail
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content

HTTP Code	Description	Schema
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `*/*`

Tags

- `order-details-controller`

Deletes an Order detail from the system. 404 if the order detail's identifier is not found.

```
DELETE /order-details/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
204	No Content	No Content
401	Unauthorized	No Content
403	Forbidden	No Content

Produces

- `application/json`

Tags

- order-details-controller

Creates a new Product.

POST /products

Parameters

Type	Name	Description	Schema
Body	product <i>required</i>	product	Product

Responses

HTTP Code	Description	Schema
200	OK	Product
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- [application/json](#)

Produces

- [*/*](#)

Tags

- product-controller

Returns list of all Products in the system.

GET /products

Responses

HTTP Code	Description	Schema
200	OK	< Product > array
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- product-controller

Returns list of Products in Warehouses where User belongs, specified by User identifier. 404 if does not exist.

GET /products/user/{id}

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	< Product > array

HTTP Code	Description	Schema
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- product-controller

Returns list of Products in Warehouse by Warehouse identifier. 404 if does not exist.

```
GET /products/warehouse/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	< Product > array
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- product-controller

Returns a specific Product by their identifier. 404 if does not exist.

```
GET /products/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	Product
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- product-controller

Updates Product.

```
PUT /products/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)
Body	product <i>required</i>	product	Product

Responses

HTTP Code	Description	Schema
200	OK	Product
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `*/*`

Tags

- product-controller

Deletes a Product from the system. 404 if the product's identifier is not found.

```
DELETE /products/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
204	No Content	No Content
401	Unauthorized	No Content
403	Forbidden	No Content

Produces

- `application/json`

Tags

- product-controller

Returns Supplier of a specific Product by their identifier. 404 if does not exist.

```
GET /supplier/product/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity

HTTP Code	Description	Schema
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- product-controller

Returns Supplier of a specific User by their identifier. 404 if does not exist.

```
GET /supplier/user/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- `application/json`

Tags

- user-controller

Creates a new Supplier.

POST /suppliers

Parameters

Type	Name	Description	Schema
Body	supplier <i>required</i>	supplier	Supplier

Responses

HTTP Code	Description	Schema
200	OK	Supplier
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `*/*`

Tags

- supplier-controller

Returns list of all Suppliers in the system.

GET /suppliers

Responses

HTTP Code	Description	Schema
200	OK	< Supplier > array
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

• */*

Tags

• supplier-controller

Returns a specific Supplier by User identifier. 404 if does not exist.

GET /suppliers/user/{id}

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	Supplier

HTTP Code	Description	Schema
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- supplier-controller

Returns a specific Supplier by their identifier. 404 if does not exist.

```
GET /suppliers/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	Supplier
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- supplier-controller

Updates Supplier.

PUT /suppliers/{id}

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)
Body	supplier <i>required</i>	supplier	Supplier

Responses

HTTP Code	Description	Schema
200	OK	Supplier
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- application/json

Produces

- */*

Tags

- supplier-controller

Deletes a Supplier from the system. 404 if the supplier's identifier is not found.

```
DELETE /suppliers/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
204	No Content	No Content
401	Unauthorized	No Content
403	Forbidden	No Content

Produces

- `application/json`

Tags

- supplier-controller

Updates User.

```
PUT /updateUser/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)
Body	user <i>required</i>	user	User

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- user-controller

Updates Warehouse.

```
PUT /updateWarehouse/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)
Body	warehouse <i>required</i>	warehouse	Warehouse

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- warehouse-controller

Creates a new User.

POST /user

Parameters

Type	Name	Description	Schema
Body	newUser <i>required</i>	newUser	User

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- user-controller

Returns list of all Users in the system.

GET /users

Responses

HTTP Code	Description	Schema
200	OK	< User > array
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- `application/json`

Tags

- user-controller

Returns a specific User by their username. 404 if does not exist.

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

Parameters

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

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- `application/json`

Tags

- user-controller

Returns a specific User by their identifier. 404 if does not exist.

GET /users/{id}

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- `application/json`

Tags

- user-controller

Creates a new Warehouse.

POST /warehouse

Parameters

Type	Name	Description	Schema
Body	newWarehouse <i>required</i>	newWarehouse	Warehouse

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- warehouse-controller

Returns Warehouse of a specific Product by their identifier. 404 if does not exist.

```
GET /warehouse/product/{id}
```

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- `*/*`

Tags

- product-controller

Returns list of all Warehouses in the system.

GET /warehouses

Responses

HTTP Code	Description	Schema
200	OK	< Warehouse > array
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- `application/json`

Tags

- warehouse-controller

Returns a specific Warehouse by their identifier. 404 if does not exist.

GET /warehouses/{id}

Parameters

Type	Name	Description	Schema
Path	id <i>required</i>	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- `application/json`

Tags

- warehouse-controller

Definitions

Category

Name	Schema
categoryName <i>optional</i>	string

Name	Schema
id <i>optional</i>	integer (int64)

OrderDetail

Name	Schema
date <i>optional</i>	string (date-time)
id <i>optional</i>	integer (int64)
productId <i>optional</i>	Product
quantity <i>optional</i>	integer (int32)
size <i>optional</i>	integer (int32)
total <i>optional</i>	number (double)
unitPrice <i>optional</i>	number (double)

Product

Name	Schema
categoryId <i>optional</i>	Category
description <i>optional</i>	string
id <i>optional</i>	integer (int64)

Name	Schema
imageUrl <i>optional</i>	string
name <i>optional</i>	string
orderDetails <i>optional</i>	< OrderDetail > array
price <i>optional</i>	number (double)
quantity <i>optional</i>	integer (int32)
status <i>optional</i>	string
supplierId <i>optional</i>	Supplier
unit <i>optional</i>	string
warehouseId <i>optional</i>	Warehouse

ResponseEntity

Name	Schema
body <i>optional</i>	object

Name	Schema
statusCode <i>optional</i>	enum (100 CONTINUE, 101 SWITCHING_PROTOCOLS, 102 PROCESSING, 103 CHECKPOINT, 200 OK, 201 CREATED, 202 ACCEPTED, 203 NON_AUTHORITATIVE_INFORMATION, 204 NO_CONTENT, 205 RESET_CONTENT, 206 PARTIAL_CONTENT, 207 MULTI_STATUS, 208 ALREADY_REPORTED, 226 IM_USED, 300 MULTIPLE_CHOICES, 301 MOVED_PERMANENTLY, 302 FOUND, 302 MOVED_TEMPORARILY, 303 SEE_OTHER, 304 NOT_MODIFIED, 305 USE_PROXY, 307 TEMPORARY_REDIRECT, 308 PERMANENT_REDIRECT, 400 BAD_REQUEST, 401 UNAUTHORIZED, 402 PAYMENT_REQUIRED, 403 FORBIDDEN, 404 NOT_FOUND, 405 METHOD_NOT_ALLOWED, 406 NOT_ACCEPTABLE, 407 PROXY_AUTHENTICATION_REQUIRED, 408 REQUEST_TIMEOUT, 409 CONFLICT, 410 GONE, 411 LENGTH_REQUIRED, 412 PRECONDITION_FAILED, 413 PAYLOAD_TOO_LARGE, 413 REQUEST_ENTITY_TOO_LARGE, 414 URI_TOO_LONG, 414 REQUEST_URI_TOO_LONG, 415 UNSUPPORTED_MEDIA_TYPE, 416 REQUESTED_RANGE_NOT_SATISFIABLE, 417 EXPECTATION_FAILED, 418 I_AM_A_TEAPOT, 419 INSUFFICIENT_SPACE_ON_RESOURCE, 420 METHOD_FAILURE, 421 DESTINATION_LOCKED, 422 UNPROCESSABLE_ENTITY, 423 LOCKED, 424 FAILED_DEPENDENCY, 425 TOO_EARLY, 426 UPGRADE_REQUIRED, 428 PRECONDITION_REQUIRED, 429 TOO_MANY_REQUESTS, 431 REQUEST_HEADER_FIELDS_TOO_LARGE, 451 UNAVAILABLE_FOR_LEGAL_REASONS, 500 INTERNAL_SERVER_ERROR, 501 NOT_IMPLEMENTED, 502 BAD_GATEWAY, 503 SERVICE_UNAVAILABLE, 504 GATEWAY_TIMEOUT, 505 HTTP_VERSION_NOT_SUPPORTED, 506 VARIANT_ALSO_NEGOTIATES, 507 INSUFFICIENT_STORAGE, 508 LOOP_DETECTED, 509 BANDWIDTH_LIMIT_EXCEEDED, 510 NOT_EXTENDED, 511 NETWORK_AUTHENTICATION_REQUIRED)
statusCodeValue <i>optional</i>	integer (int32)

Supplier

Name	Schema
address <i>optional</i>	string
email <i>optional</i>	string
fax <i>optional</i>	string
id <i>optional</i>	integer (int64)
name <i>optional</i>	string
otherDetails <i>optional</i>	string
phone <i>optional</i>	string
users <i>optional</i>	< User > array

User

Name	Schema
first_name <i>optional</i>	string
id <i>optional</i>	integer (int64)
last_name <i>optional</i>	string
supplierID <i>optional</i>	Supplier

Name	Schema
username <i>optional</i>	string
warehouseID <i>optional</i>	Warehouse

Warehouse

Name	Schema
company_name <i>optional</i>	string
id <i>optional</i>	integer (int64)
products <i>optional</i>	< Product > array
users <i>optional</i>	< User > array