



API

1.0

OAS 3.0

Restaurant Management System API

API

default

GET

/

categories

POST

/categories

Parameters

Try it out

No parameters

Request body

required

application/json

Example Value

Schema

```
{
  "name": "string",
  "slug": "string",
  "description": "string"
}
```

Responses

Code

Description

Links

201

No links

GET

/categories

^

Parameters

Try it out

No parameters

Responses

Code	Description	Links
200		No links

GET

/categories/{slug}

^

Parameters

Try it out

Name	Description
slug * required string (path)	<div>slug</div>

Responses

Code	Description	Links
200		No links

PATCH

/categories/{id}

^

Parameters

Try it out

Name	Description
id * required string	<div>id</div>

NameDescription

(path)

Request body required

application/json

Example Value | Schema

```
{  "name": "string",  "slug": "string",  "description": "string"}  
```

Responses

Code	Description	Links
200		No links

products

POST /products

Parameters

Try it out

No parameters

Request body required

application/json

Example Value | Schema

```
{  "name": "string",  "description": "string",  "categoryId": "string",  "price": 0,  "quantity": 0,  "status": "string"}  
```

Responses

Code	Description	Links
201		No links

GET /products ^

Parameters

Try it out

No parameters

Responses

Code	Description	Links
200		No links

PATCH /products/{id} ^

Parameters

Try it out

Name	Description
id * required	
string	id
(path)	

Request body required

application/json v

Example Value | Schema

```
{  "name": "string",  "description": "string",  "categoryId": "string",  "price": 0,  "quantity": 0,  "status": "string"}}
```

Responses

Code	Description	Links
200		No links

DELETE

/products/{id}

Parameters

Try it out

Name	Description
id * required string (path)	<div>id</div>

Responses

Code	Description	Links
200		No links

customers

POST

/customers

Parameters

Try it out

No parameters

Request body required

application/json

Example Value | Schema

```
{
  "name": "string",
  "email": "string",
  "cpf": "string"
}
```

Responses		
Code	Description	Links
201		No links

GET

/customers

^

Parameters

Try it out

No parameters

Responses

Code	Description	Links
200		No links

GET

/customers/{cpf}

^

Parameters

Try it out

Name	Description
cpf * required string (path)	<div>cpf</div>

Responses

Code	Description	Links
200		No links

PATCH

/customers/{cpf}

⌵

Parameters

Try it out

Name	Description
cpf * required string (path)	<div>cpf</div>

Request body

required

application/json ⌵

Example Value

Schema

```
{
  "name": "string",
  "email": "string",
  "cpf": "string"
}
```

Responses

Code	Description	Links
200		No links

DELETE

/customers/{cpf}

⌵

Parameters

Try it out

Name	Description
cpf * required string (path)	<div>cpf</div>

Responses

Code	Description	Links
200		No links

orders



POST

/orders

Parameters

Try it out

No parameters

Request body required

application/json

Example Value

Schema

```
{
  "customer_cpf": "string",
  "product_id_amount_id": {
    "id_produto_1": 2,
    "id_produto_2": 5
  }
}
```

Responses

Code	Description	Links
201		No links

GET

/orders

Parameters

Try it out

No parameters

Responses

Code	Description	Links
200		No links

Schemas



CreateCategoryDto ▾ {
 name* string
 slug* string
 description* string
}

UpdateCategoryDto ▾ {
 name string
 slug string
 description string
}

CreateProductDto ▾ {
 name* string
 description* string
 categoryId* string
 price* number
 quantity* number
 status* string
}

UpdateProductDto ▾ {
 name string
 description string
 categoryId string
 price number
 quantity number
 status string
}

CreateCustomerDto ▾ {
 name* string
 email* string
 cpf* string
}

UpdateCustomerDto ▾ {
 name string
 email string
 cpf string
}

Map ▾ {
}

CreateOrderDto ▾ {
 customer_cpf* string
 product_id_amount_id* ▾ {
 }
 example: OrderedMap { "id_produto_1": 2, "id_produto_2": 5 }
}