ExpensesManager

Table of Contents

Chapter 1. Overview

ExpensesManager is a service to expenses management.

1.1. Version information

Version : 1.0.0

1.2. License information

License : Proprietary
Terms of service : null

1.3. URI scheme

Host : localhost:8080
BasePath : /
Schemes : HTTPS

1.4. Tags

- BillOfSale Command
- Budget
- Order Command
- Product Command
- Product Query

Chapter 2. Resources

2.1. BillOfSale Command

2.1.1. Create bill of sale.

```
POST /bills
```

Description

Method allow to create a new bill of sale.

Responses

HTTP Code	Description	Schema
201	BillOfSale was created.	BillOfSale
409	BillOfSale bought date cannot be null.	No Content

Example HTTP request

Request path

```
/bills
```

Example HTTP response

Response 201

```
{
  "billId" : {
    "id" : "string"
  },
  "ordersList" : [ {
    "orderId" : {
      "id" : "string"
    },
    "product" : {
      "productId" : {
        "id" : "string"
      },
      "name" : "string",
      "price" : 0.0
    },
    "quanity" : 0
  } ],
  "boughtDate" : 0,
  "description" : "string"
}
```

2.1.2. Remove bill of sale.

```
DELETE /bills/{id}
```

Description

Method allow to remove a bill of sale.

Responses

HTTP Code	Description	Schema
204	BillOfSale was removed.	No Content
409	ID has invalid format.	No Content

Example HTTP request

Request path

```
/bills/{id}
```

2.2. Budget

2.2.1. Create budget.

```
POST /budgets
```

Description

Method allow to create a new budget.

Responses

HTTP Code	Description	Schema
201	Budget was created.	Budget
409	Budget value cannot be null or is infinite.	No Content

Example HTTP request

Request path

```
/budgets
```

Example HTTP response

Response 201

```
{
  "budgetId" : {
    "id" : "string"
  },
  "name" : "string",
  "budgetValue" : 0.0,
  "billsOfSalseList" : [ {
    "billId" : {
      "id" : "string"
    },
    "ordersList" : [ {
      "orderId" : {
        "id" : "string"
      },
      "product" : {
        "productId" : {
          "id" : "string"
        },
        "name" : "string",
        "price" : 0.0
      },
      "quanity" : 0
    } ],
    "boughtDate" : 0,
    "description" : "string"
  } ]
}
```

2.2.2. Remove budget.

```
DELETE /budgets/{id}
```

Description

Method allow to remove a budget.

Responses

HTTP Code	Description	Schema
204	Budget was removed.	No Content
409	ID has invalid format.	No Content

Example HTTP request

Request path

```
/budgets/{id}
```

2.3. Order Command

2.3.1. Create order.

```
POST /orders
```

Description

Method allow to create a new order.

Responses

HTTP Code	Description	Schema
201	Order was created.	ResponseNewOrder
409	Order quanity cannot be null or is infinite.	No Content

Example HTTP request

Request path

```
/orders
```

Example HTTP response

Response 201

```
{
  "id" : "string",
  "productId" : "string",
  "quanity" : 0
}
```

2.3.2. Remove order.

```
DELETE /orders/{id}
```

Description

Method allow to remove an order by id.

Responses

HTTP Code	Description	Schema
204	Order was removed.	No Content
409	ID has invalid format.	No Content

Example HTTP request

Request path

```
/orders/{id}
```

2.4. Product Command

2.4.1. Create product.

```
POST /products
```

Description

Method allow to create a new product.

Responses

HTTP Code	Description	Schema
201	Product was created.	ResponseNewProduct
409	Product price cannot be null or is infinite.	No Content

Example HTTP request

Request path

```
/products
```

Example HTTP response

Response 201

```
{
  "id" : "string",
  "name" : "string",
  "price" : 0.0
}
```

2.4.2. Remove product.

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

Description

Method allow to remove a product by id.

Parameters

Type	Name	Schema
Path	**id** *required*	string

Responses

HTTP Code	Description	Schema
204	Product was removed.	No Content
409	ID has invalid format.	No Content

Example HTTP request

Request path

```
/products/string
```

2.5. Product Query

2.5.1. Find product by id.

```
GET /products/id
```

Description

Method allow to find a product by id.

Responses

HTTP Code	Description	Schema
200	Product was found.	Product
404	Product not found.	No Content
409	ID has invalid format.	No Content

Example HTTP request

Request path

```
/products/id
```

Example HTTP response

Response 200

```
{
  "productId" : {
    "id" : "string"
  },
  "name" : "string",
  "price" : 0.0
}
```

Chapter 3. Definitions

3.1. BillOfSale

Name	Description	Schema
billId *optional*	**Example** : DomainID	DomainID
boughtDate *optional*	**Example** : 0	integer (int64)
description *optional*	**Example** : "string"	string
ordersList *optional*	**Example** : ["Order"]	< Order > array

3.2. Budget

Name	Description	Schema
billsOfSalseList *optional*	**Example** : ["BillOfSale"]	< BillOfSale > array
budgetId *optional*	**Example** : DomainID	DomainID
budgetValue *optional*	**Example** : 0.0	number (double)
name *optional*	**Example** : "string"	string

3.3. DomainID

Name	Description	Schema
id *optional*	**Example** : "string"	string

3.4. Order

Name	Description	Schema
orderId *optional*	**Example** : DomainID	DomainID
product *optional*	**Example** : Product	Product
quanity *optional*	**Example** : `0`	integer (int32)

3.5. Product

Name	Description	Schema
name *optional*	**Example** : `"string"`	string
price *optional*	**Example** : `0.0`	number (double)
productId *optional*	**Example** : DomainID	DomainID

3.6. ResponseNewOrder

Name	Description	Schema
id *optional*	**Example** : `"string"`	string
productId *optional*	**Example** : `"string"`	string
quanity *optional*	**Example** : `0`	integer (int32)

3.7. ResponseNewProduct

Name	Description	Schema
id *optional*	**Example** : `"string"`	string
name *optional*	**Example** : `"string"`	string
price *optional*	**Example** : `0.0`	number (double)