Contents

[Purpose of this document 2](#_Toc135087953)

[REST resources (services) 3](#_Toc135087954)

[Products 3](#_Toc135087955)

[Manufacturers 3](#_Toc135087956)

[Customers 3](#_Toc135087957)

[Orders 3](#_Toc135087958)

[REST requests and responses 4](#_Toc135087959)

[1 Products 4](#_Toc135087960)

[1.1 Get all products 4](#_Toc135087961)

[1.2 Insert new product 5](#_Toc135087962)

[1.3 Edit product 5](#_Toc135087963)

[1.4 Delete product 6](#_Toc135087964)

[2 Manufacturers 6](#_Toc135087965)

[2.1 Get all manufacturers 6](#_Toc135087966)

[2.2 Insert new manufacturer 7](#_Toc135087967)

[2.3 Edit manufacturer 7](#_Toc135087968)

[2.4 Delete manufacturer 7](#_Toc135087969)

[3 Customers 8](#_Toc135087970)

[3.1 Get all customers 8](#_Toc135087971)

[3.2 Insert new customer 8](#_Toc135087972)

[3.3 Edit customer 9](#_Toc135087973)

[3.4 Delete customer 9](#_Toc135087974)

[4 Orders 9](#_Toc135087975)

[4.1 Get all orders 9](#_Toc135087976)

[4.2 Insert new order 10](#_Toc135087977)

[4.3 Edit order 11](#_Toc135087978)

[4.4 Delete order 13](#_Toc135087979)

[5 HTTP methods 13](#_Toc135087980)

# Purpose of this document

This document describes what REST services Omppu & Rane -application provides. It contains all needed information so that client can be implemented.

# REST resources (services)

## Products

1. Get all products -> http://localhost:8080/api/products (use plurals)

2. Insert new product -> http://localhost:8080/api/products

3. Edit product -> http://localhost:8080/api/products/{id} (id parameter needed)

4. Delete product -> http://localhost:8080/api/products/{id}

## Manufacturers

1. Get all manufacturers -> http://localhost:8080/api/manufacturers (use plurals)

2. Insert new manufacturer -> http://localhost:8080/api/manufacturers

3. Edit manufacturer -> http://localhost:8080/api/manufacturers/{id} (id parameter needed)

4. Delete manufacturer -> http://localhost:8080/api/manufacturers/{id}

## Customers

1. Get all customers -> http://localhost:8080/api/customers (use plurals)

2. Insert new customer -> http://localhost:8080/api/customers

3. Edit customer -> http://localhost:8080/api/customers/{id} (id parameter needed)

4. Delete customer -> http://localhost:8080/api/customers/{id}

## Orders

1. Get all orders -> http://localhost:8080/api/orderinfoes (use plurals)

2. Insert new order -> http://localhost:8080/api/orderinfoes

3. Edit order -> http://localhost:8080/api/orderinfoes/{id} (id parameter needed)

4. Delete order -> http://localhost:8080/api/orderinfoes/{id}

# REST requests and responses

## 1 Products

### 1.1 Get all products

HTTP method: GET

API call: http://localhost:8080/api/products

Request body: -

Response example below:

[

    {

        "id": 1,

        "name": "Sadetakki",

"type": {

            "id": 2,

            "name": "Vaate"

        },

"size": {

            "id": 2,

            "clothingSize": "M"

        },

        "price": 39.9,

        "quantity": 3,

        "manufacturer": {

            "id": 1,

            "name": "Purina"

        }

    },

    {

        "id": 2,

        "name": "Ulkotossut",

"type": {

            "id": 2,

            "name": "Vaate"

        },

"size": {

            "id": 1,

            "clothingSize": "S"

        },

        "price": 19.9,

        "quantity": 6,

        "manufacturer": {

            "id": 2,

            "name": "Pedigree"

        }

    },

    {

        "id": 3,

        "name": "Hassuhattu",

"type": {

            "id": 2,

            "name": "Vaate"

        },

"size": {

            "id": 3,

            "clothingSize": "L"

        },

        "price": 12.9,

        "quantity": 5,

        "manufacturer": {

            "id": 1,

            "name": "Purina"

        }

    }

]

### 1.2 Insert new product

HTTP method: POST

API call: http://localhost:8080/api/products

Request body:

{

    "name": "Köysilelu",

"type": {

        "id": 1,

        "name": "Lelu"

    },

"size": {

        "id": 4,

        "clothingSize": "-"

    },

    "price": 6.9,

    "quantity": 1,

    "manufacturer": {

        "id": 1,

        "name": "Purina"

    }

}

Response example below (Status 200 OK):

{

    "id": 4,

    "name": "Köysilelu",

"type": {

        "id": 1,

        "name": "Lelu"

    },

"size": {

        "id": 4,

        "clothingSize": "-"

    },

    "price": 6.9,

    "quantity": 1,

    "manufacturer": {

        "id": 1,

        "name": "Purina"

    }

}

### 1.3 Edit product

HTTP method: PUT

API call: http://localhost:8080/api/products/1

Request body:

{

    "name": "Sadetakki updated",

"type": {

        "id": 2,

        "name": "Vaate"

    },

"size": {

        "id": 2,

        "clothingSize": "M"

    },

    "price": 39.9,

    "quantity": 3,

    "manufacturer": {

        "id": 1,

        "name": "Purina"

    }

}

Response example below (Status 200 OK):

{

    "id": 1,

    "name": "Sadetakki updated",

"type": {

        "id": 2,

        "name": "Vaate"

    },

"size": {

        "id": 2,

        "clothingSize": "M"

    },

    "price": 39.9,

    "quantity": 3,

    "manufacturer": {

        "id": 1,

        "name": "Purina"

    }

}

### 1.4 Delete product

HTTP method: DELETE

API call: http://localhost:8080/api/product/4

Request body: -

Response: Status 200 OK

## 2 Manufacturers

### 2.1 Get all manufacturers

HTTP method: GET

API call: http://localhost:8080/api/manufacturers

Request body: -

Response example below:

[

    {

        "id": 1,

        "name": "Purina"

    },

    {

        "id": 2,

        "name": "Pedigree"

    }

]

### 2.2 Insert new manufacturer

HTTP method: POST

API call: http://localhost:8080/api/manifacturers

Request body:

{

    "name": "Rukka"

}

Response example below (Status 200 OK):

{

    "id": 3,

    "name": "Rukka"

}

### 2.3 Edit manufacturer

HTTP method: PUT

API call: http://localhost:8080/api/manufacturers/3

Request body:

{

    "type": "Rukka"

}

Response example below (Status 200 OK):

{

    "id": 3,

    "name": "Rukka"

}

### 2.4 Delete manufacturer

HTTP method: DELETE

API call: http://localhost:8080/api/manufacturers/3

Request body: -

Response: Status 200 OK

## 3 Customers

### 3.1 Get all customers

HTTP method: GET

API call: http://localhost:8080/api/customers

Request body: -

Response example below:

[

    {

        "id": 1,

        "firstName": "Riku",

        "lastName": "Rekka",

        "phone": "0409620451",

        "email": "riku@email.com",

    },

    {

        "id": 2,

        "firstName": "Saku",

        "lastName": "Sauna",

        "phone": "0401234551",

        "email": "saku@email.com",

    }

]

### 3.2 Insert new customer

HTTP method: POST

API call: http://localhost:8080/api/customers

Request body:

{

    "firstName": "Maija",

    "lastName": "Mattila",

    "phone": "0401238900",

    "email": "maija@email.com",

}

Response example below (Status 200 OK):

{

    "id": 3,

    "firstName": "Maija",

    "lastName": "Mattila",

    "phone": "0401238900",

    "email": "maija@email.com",

}

### 3.3 Edit customer

HTTP method: PUT

API call: http://localhost:8080/api/customers/4

Request body:

{

    "firstName": "Maija",

    "lastName": "Mattila",

    "phone": "0401238900",

    "email": "maija1@email.com"

}

Response example below (Status 200 OK):

{

    "id": 3,

    "firstName": "Maija",

    "lastName": "Mattila",

    "phone": "0401238900",

    "email": "maija1@email.com"

}

### 3.4 Delete customer

HTTP method: DELETE

API call: http://localhost:8080/api/customers/3

Request body: -

Response: Status 200 OK

## 4 Orders

### 4.1 Get all orders

HTTP method: GET

API call: http://localhost:8080/api/orderinfoes

Request body: -

Response example below:

[

    {

        "id": 1,

"date": "2023-05-15",

        "customer": {

            "id": 4,

            "firstName": "Maija",

            "lastName": "Mattila",

            "phone": "0401238900",

            "email": "maija1@email.com"

        },

        "product": {

            "id": 1,

            "name": "Sadetakki",

            "type": {

                "id": 2,

                "name": "Vaate"

            },

            "size": {

                "id": 2,

                "clothingSize": "M"

            },

            "price": 39.9,

            "quantity": 3,

            "manufacturer": {

                "id": 1,

                "name": "Purina"

                }

        }

        "orderstatus": "Tilaus toimitettu"

    }

]

### 4.2 Insert new order

HTTP method: POST

API call: http://localhost:8080/api/orderinfoes

Request body:

{

    "date": "2023-05-17",

"customer": {

        "id": 3,

        "firstName": "Maija",

        "lastName": "Mattila"

        "phone": "0401238900",

        "email": [maija@email.com](mailto:maija@email.com)

    },

    "product": {

        "id": 1,

        "name": "Sadetakki",

        "type": {

            "id": 2,

            "name": "Vaate"

        },

        "size": {

            "id": 2,

            "clothingSize": "M"

        },

        "price": 39.9,

        "quantity": 3,

        "manufacturer": {

            "id": 1,

            "name": "Purina"

        }

    }

"orderstatus": "Tilaus vastaanotettu"

}

Response example below (Status 200 OK):

{

    "id": 2,

"date": "2023-05-17",

    "customer": {

        "id": 3,

        "firstName": "Maija",

        "lastName": "Mattila"

        "phone": "0401238900",

        "email": [maija@email.com](mailto:maija@email.com)

    },

    "product": {

        "id": 1,

        "name": "Sadetakki",

        "type": {

            "id": 2,

            "name": "Vaate"

        },

        "size": {

            "id": 2,

            "clothingSize": "M"

        },

        "price": 39.9,

        "quantity": 3,

        "manufacturer": {

            "id": 1,

            "name": "Purina"

        }

    }

"orderstatus": "Tilaus vastaanotettu"

}

### 4.3 Edit order

HTTP method: PUT

API call: http://localhost:8080/api/orderinfoes /3

Request body:

{

    "id": 2,

    "date": "2023-05-17",

    "customer": {

        "id": 4,

        "firstName": "Maija",

        "lastName": "Mattila"

        "phone": "0401238900",

        "email": [maija1@email.com](mailto:maija1@email.com)

    },

    "product": {

        "id": 1,

        "name": "Sadetakki",

        "type": {

            "id": 2,

            "name": "Vaate"

        },

        "size": {

            "id": 2,

            "clothingSize": "M"

        },

        "price": 39.9,

        "quantity": 3,

        "manufacturer": {

            "id": 1,

            "name": "Purina"

        }

    }

    "orderstatus": "Tilaus peruutettu"

}

Response example below (Status 200 OK):

{

    "id": 2,

    "date": "2023-05-17",

    "customer": {

        "id": 4,

        "firstName": "Maija",

        "lastName": "Mattila"

        "phone": "0401238900",

        "email": [maija1@email.com](mailto:maija1@email.com)

    },

    "product": {

        "id": 1,

        "name": "Sadetakki",

        "type": {

            "id": 2,

            "name": "Vaate"

        },

        "size": {

            "id": 2,

            "clothingSize": "M"

        },

        "price": 39.9,

        "quantity": 3,

        "manufacturer": {

            "id": 1,

            "name": "Purina"

        }

    }

    "orderstatus": "Tilaus peruutettu"

}

### 4.4 Delete order

HTTP method: DELETE

API call: http://localhost:8080/api/orderinfoes/1

Request body: -

Response: Status 200 OK

## 5 HTTP methods

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Http**  **Method** | **Resource EndPoint** | **Input** | **Success Response** | **Error**  **Response** | **Description** |
| GET | /api/products | Body: empty | Status: 200  List of products | Status: 500 | Retrieves all products |
| GET | /api/products/id/{id} | Body: empty | Status: 200  Information of one product | Status: 500 | Retrieves information of one product |
| GET | /api/products/type/{type} | Body: empty | Status: 200  List of products | Status: 500 | Retrieves list of products with the same type |
| POST | /api/products | Body: new product data | Status: 200  New product with id information | Status: 500 | Creates new product |
| PUT | /api/ products /{id} | Body: product data with updates | Status: 200  Body: Empty | Status: 404 or 500 | Updates an existing product |
| DELETE | /api/ products/{id} | Body: empty | Status: 200 | Status: 404 or 500 | Deletes an existing product |
| GET | /api/manufacturers | Body: empty | Status: 200  List of manufacturers | Status: 500 | Retrieves all manufacturers |
| GET | /api/manufacturers/{id} | Body: empty | Status: 200  Information of one manufacturer | Status: 500 | Retrieves information of one manufacturer |
| POST | /api/manufacturers | Body: new manufacturer data | Status: 200  New manufacturer with id information | Status: 500 | Creates new manufacturer |
| PUT | /api/manufacturers/{id} | Body: manufacturer data with updates | Status: 200  Body: Empty | Status: 404 or 500 | Updates an existing manufacturer |
| DELETE | /api/manufacturers/{id} | Body: empty | Status: 200 | Status: 404 or 500 | Deletes an existing manufacturer |
| GET | /api/customer | Body: empty | Status: 200  List of customers | Status: 500 | Retrieves all customers |
| GET | /api/customer/{id} | Body: empty | Status: 200  Information of one customer | Status: 500 | Retrieves information of one customer |
| POST | /api/customer | Body: new customer data | Status: 200  New customer with id information | Status: 500 | Creates new customer |
| PUT | /api/customer/{id} | Body: customer data with updates | Status: 200  Body: Empty | Status: 404 or 500 | Updates an existing customer |
| DELETE | /api/customer/{id} | Body: empty | Status: 200 | Status: 404 or 500 | Deletes an existing customer |
| GET | /api/orderinfoes | Body: empty | Status: 200  List of orders | Status: 500 | Retrieves all orders |
| GET | /api/orderinfoes/{id} | Body: empty | Status: 200  Information of one order | Status: 500 | Retrieves information of one order |
| POST | /api/orderinfoes | Body: new orderinfo data | Status: 200  New order with id information | Status: 500 | Creates new order |
| PUT | /api/orderinfoes/{id} | Body: orderinfo data with updates | Status: 200  Body: Empty | Status: 404 or 500 | Updates an existing order |
| DELETE | /api/orderinfoes/{id} | Body: empty | Status: 200 | Status: 404 or 500 | Deletes an existing order |