Shopping Cart API Documentation

Base URL: `https://localhost:7140/api`

This document describes the REST APIs for managing **Products, Clients, Carts, and Cart Items** in the Shopping Cart system.

---

1. Products API

1.1 Create Product

Endpoint: `POST /Products`

Request Body

```json
{
  "name": "string",
  "type": "HighEndPhone",
  "individualPrice": 10,
  "professionalHighRevenuePrice": 10,
  "professionalLowRevenuePrice": 10
}
```

Response (201 OK)

```json
{
  "id": 5,
  "name": "string",
  "type": "HighEndPhone",
  "individualPrice": 10,
  "professionalHighRevenuePrice": 10,
  "professionalLowRevenuePrice": 10,
  "cartItems": []
}
```

1.2 Get All Products

Endpoint: `GET /Products`

Response (200 OK)

```
[
  {
    "id": 1,
    "name": "High-End Phone",
    "type": "HighEndPhone",
    "individualPrice": 1500,
    "professionalHighRevenuePrice": 1000,
    "professionalLowRevenuePrice": 1150,
    "cartItems": []
  },
  {
    "id": 2,
    "name": "Mid-Range Phone",
    "type": "MidRangePhone",
    "individualPrice": 800,
    "professionalHighRevenuePrice": 550,
    "professionalLowRevenuePrice": 600,
    "cartItems": []
  }
]
```

1.3 Get Product by ID

Endpoint: `GET /Products/{id}`

Response (200 OK)

```
{
  "id": 1,
  "name": "High-End Phone",
  "type": "HighEndPhone",
  "individualPrice": 1500,
  "professionalHighRevenuePrice": 1000,
  "professionalLowRevenuePrice": 1150,
  "cartItems": []
}
```

1.4 Update Product

Endpoint: `PUT /Products/{id}`

Request Body

```
{
  "id": 5,
  "name": "Test",
  "type": "HighEndPhone",
  "individualPrice": 100,
  "professionalHighRevenuePrice": 1200,
  "professionalLowRevenuePrice": 130
}
```

Response (200 OK)

```
{
  "id": 5,
  "name": "Test",
  "type": "HighEndPhone",
  "individualPrice": 100,
  "professionalHighRevenuePrice": 1200,
  "professionalLowRevenuePrice": 130,
  "cartItems": []
}
```

1.5 Delete Product

Endpoint: `DELETE /Products/{id}`

Response (200 OK)

```
true
```

---

2. Clients API

2.1 Create Client

Professional Client

Endpoint: `POST /Clients`

Request Body

```json
{
  "clientType": "Professional",
  "companyName": "string",
  "vatNumber": "string",
  "businessRegNo": "string",
  "annualRevenue": 100000000
}
```

Response

```json
{
  "id": 12,
  "clientType": "Professional",
  "firstName": null,
  "lastName": null,
  "companyName": "string",
  "vatNumber": "string",
  "businessRegNo": "string",
  "annualRevenue": 100000000,
  "carts": []
}
```

Individual Client

```json
{
  "clientType": "Individual",
  "firstName": "string",
  "lastName": "string"
}
```

Response

```json
{
  "id": 13,
  "clientType": "Individual",
  "firstName": "string",
  "lastName": "string",
  "companyName": null,
  "vatNumber": null,
  "businessRegNo": null,
  "annualRevenue": null,
  "carts": []
}
```

2.2 Update Client

Endpoint: `PUT /Clients/{id}`

Individual Client Request

```
{
  "id": 13,
  "clientType": "Individual",
  "firstName": "Ramakrishnan",
  "lastName": "Ramar"
}
```

Professional Client Request

```
{
  "id": 12,
  "clientType": "Professional",
  "companyName": "Decent Designers",
  "vatNumber": "BE290494",
  "businessRegNo": "29094",
  "annualRevenue": 100000000
}
```

2.3 Get All Clients

Endpoint: `GET /Clients`

Response (200 OK)

```
[
  {
    "id": 13,
    "clientType": "Individual",
    "firstName": "Ramakrishnan",
    "lastName": "Ramar",
    "companyName": null,
    "vatNumber": null,
    "businessRegNo": null,
    "annualRevenue": 0,
    "carts": []
  }
]
```

2.4 Get Client by ID

Endpoint: `GET /Clients/{id}`

2.5 Delete Client

Endpoint: `DELETE /Clients/{id}`

Response

```
true
```

---

3. Carts API

3.1 Create Cart

Endpoint: `POST /Carts`

Request Body

```
{
  "clientId": 11
}
```

Response

```
{
  "id": 6,
  "clientId": 11,
  "total": 0,
  "items": []
}
```

3.2 Update Cart

Endpoint: `PUT /Carts/{id}`

3.3 Get All Carts

Endpoint: `GET /Carts`

3.4 Get Cart by ID

Endpoint: `GET /Carts/{id}`

3.5 Delete Cart

Endpoint: `DELETE /Carts/{id}`

Response

```
true
```

4. Cart Items API

4.1 Add / Update Cart Items

Endpoint: `POST /CartItems`

Request Body

```
{
  "id": 5,
  "items": [
    { "productId": 1, "quantity": 1 },
    { "productId": 2, "quantity": 2 },
    { "productId": 3, "quantity": 3 }
  ]
}
```

Response

```
{
  "id": 5,
  "clientId": 11,
  "total": 4800,
  "items": [
    { "productId": 1, "quantity": 1, "unitPrice": 1000, "totalUnitPrice":
1000 },
    { "productId": 2, "quantity": 2, "unitPrice": 550, "totalUnitPrice": 1100 },
    { "productId": 3, "quantity": 3, "unitPrice": 900, "totalUnitPrice": 2700 }
  ]
}
```

Notes

- Prices are calculated dynamically based on **Client Type** and **Annual Revenue**.
- Professional clients receive different pricing tiers (High / Low Revenue).
- All endpoints follow RESTful conventions and return JSON responses.