

# Retail System

## Summary:

Experience the power of our Retail System Website, built on the robust foundation of .NET Core and Blazor.

Whether you are a small business or a large enterprise, our platform is tailored to meet your retail needs. Start exploring, managing, and thriving in the world of retail technology.

Version	Date	Developer	Description
0.0	20-Dec-2023	Mahmoud Khaled Ragab	Initial

# TABLE OF CONTENTS

- **Instructions**
- **Status Code**

## **1. All Categories**

- Endpoint
- Description
- Request
- Response

## **2. Add Category**

- Endpoint
- Description
- Request
- Response

## **3. Update Category**

- Endpoint
- Description
- Request
- Response

## **4. Get Category By ID**

- Endpoint
- Description
- Request
- Response

## **5. Delete Category**

- Endpoint
- Description
- Request
- Response
- 

## **6. Get All Products**

- Endpoint
- Description
- Request
- Response

## **7. Add Product**

- Endpoint
- Description
- Request
- Response

## **8. Get Product By ID**

- Endpoint
- Description
- Request
- Response

## **9. Get Name Of Product**

- Endpoint
- Description
- Request
- Response

## **10. Get Products By Category Id**

- Endpoint
- Description
- Request
- Response

## **11. Delete Product**

- Endpoint
- Description
- Request
- Response

## **12. Update Product**

- Endpoint
- Description
- Request
- Response

# Instructions

## 1. Intro

Welcome to the Retail API documentation for [Your Retail System Name]. This API allows developers to interact with various aspects of the retail system, including managing categories and products. Whether you are building a custom frontend, integrating with third-party applications, or automating tasks, this API provides the functionality you need.

## 1. Technology Used

1. Backend => .NET Core Web API
2. Frontend => Blazor.
3. Databasse => SQL Server.
4. LINQ, EF Core, Bootstrap, HTML, CSS
5. Libraries => Radzen, Syncfusion .
6. Depend On=> SOLID, Dependency inversion, Inversion of Control, RESTful APIs

## 2. Prerequisites

Before setting up and running the application, ensure you have the following prerequisites:

- .NET SDK installed on your machine.
- Visual Studio (optional but recommended for development).

### 3. Running the Application

1. Get Source Code From GitHub  
<https://github.com/mahmoudxkhaled/Inventory-Management-System.git>
2. Make sure the startup project "Retail.API".
3. Make sure migration is on "Retail.DAL". [ Add-Migration and Update-DataBase ]
4. Check your Local database and you will find "RetailBusiness" database there.
5. Make "Retail.API" and "Retail.Client" as startup project from Configure Startup projects.
6. Build and Run.

### 4. API Endpoints

Application exposes the following API endpoints:

Endpoint	HTTP Method	Description
<b>Category</b>		
`/api/category`	GET	Get List of Categories.
`/api/category`	PUT	Update an existing Category.
`/api/category/Add`	POST	Add New Category.
`/api/category/{categorytId}`	GET	Get Category By ID
`/api/category/{categorytId}`	DELETE	Delete Category
<b>Product</b>		
`/api/product`	GET	Get List of Products.
`/api/product/{productId}`	PUT	Update an existing Product.
`/api/product/Add`	POST	Add New Product.
`/api/product/{ productId }`	GET	Get Product By ID
`/api/product/{ productId }`	DELETE	Delete Product
`/api/product/GetProductsByCategoryId/{categoryId}`	GET	Get Products By Category Id
`/api/product/GetName/{productId}`	GET	Get Name of a product by Id

# Status Codes

All status codes are standard HTTP status codes. The below ones are used in this API.

- 2XX - Success of some kind
- 4XX - Error occurred in client's part
- 5XX - Error occurred in server's part

Status Code	Description
200	OK
201	Created
202	Accepted (Request accepted, and queued for execution)
400	Bad request
401	Authentication failure
403	Forbidden
404	Resource not found
405	Method Not Allowed
409	Conflict
412	Precondition Failed
500	Internal Server Error
501	Not Implemented
503	Service Unavailable

# 1. Category Controller

## 1.1. All Categories

Endpoint: **GET** /api/Category  
Description: Get all categories.  
Request: None  
Response: List of CategoryDto

## 1.2. Add Category

Endpoint: **POST** /api/Category/Add  
Description: Add a new category.  
Request: CategoryDto.  
Response: Success or BadRequest.

## 1.3. Update Category

Endpoint: **PUT** /api/Category.  
Description: Update an existing category.  
Request: CategoryDto.  
Response: Success or NotFound.

## 1.4. Get Category By ID

Endpoint: **GET** /api/Category/{categoryId}.  
Description: Get details of a specific category by ID.  
Request: None.  
Response: CategoryDto or NotFound.

## 1.5. Delete Category

Endpoint: **Delete** /api/Category/{categoryId}.  
Description: Get details of a specific category by ID.  
Request: None.  
Response: NoContent or NotFound.

## 2. Product Controller

### 2.1. All Products

Endpoint: **GET** /api/Product.  
Description: Get all products.  
Request: None.  
Response: List of ProductDto.

### 2.2. Add Product

Endpoint: **POST** /api/Product/Add.  
Description: Add a new product.  
Request: ProductDto.  
Response: Success or BadRequest

### 2.3. Update Product

Endpoint: **PUT** /api/Product.  
Description: Update an existing product.  
Request: ProductDto.  
Response: Success or NotFound.

### 2.4. Delete Product

Endpoint: **DELETE** /api/Product/{productId}.  
Description: Delete a product by ID.  
Request: None.  
Response: NoContent or NotFound

### 2.5. Get Product By ID

Endpoint: **GET** /api/Product/{productId}.  
Description: Get details of a specific product by ID.  
Request: None.  
Response: ProductDto or NotFound.

## 2.6. Get Products By Category Id

Endpoint: **GET** /api/Product/GetProdcutsByCategoryId/{categoryId}.

Description: Get a product by Category Id.

Request: None.

Response: ProductDto or NotFound.

## 2.7. Get Name

Endpoint: **GET** /api/Product/GetName/{productId}.

Description: Get Name of a product by Id

Request: None.

Response: ProductDto or NotFound

# Conclusion

- The Retail System Dashboard offers a powerful tool for managing your retail operations. If you have any questions or need assistance, please refer to the documentation or contact me.
- Feel free to customize this template based on the specific features and functionalities of your website's dashboard. Add any additional sections or details that are relevant to your users.

---

*By: Mahmoud Khaled – Software Developer*

---