

Inventory Management API Documentation

This document provides a human-readable overview of the API endpoints for the Inventory Management System.

1. Authentication Endpoints

These endpoints handle user registration and login, providing access to the system.

1.1. Register a New User

- **URL:** /register
- **Method:** POST
- **Description:** Allows a new user to create an account by providing a unique username and a password.
- **Request Body (JSON):**

```
{  "username": "newuser",  "password": "securepassword123"}
```
- **Responses:**
 - **201 Created:**

```
{  "message": "New user created!"}
```
 - **400 Bad Request:** If username or password are missing.

```
{  "message": "Username and password required!"}
```
 - **409 Conflict:** If the username already exists.

```
{  "message": "User already exists!"}
```

1.2. Log In User

- **URL:** /login
- **Method:** POST

- **Description:** Authenticates an existing user. Upon successful login, it returns a JSON Web Token (JWT) that must be used for accessing protected product endpoints.
- **Request Body (JSON):**

```
{
  "username": "existinguser",
  "password": "theirpassword"
}
```
- **Responses:**
 - **200 OK:**

```
{
  "access_token":
  "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJkdWJsaWNfaWQiOiI2NjliM2Y3YjM2YjE5YjAwNDg0N2I3YjEiLCJleHAiOiJlE2ODk2NzY4MDB9.some_jwt_token_string"
}
```
 - **401 Unauthorized:** If authentication fails (e.g., incorrect username or password, or missing credentials).
 - Could not verify (generic message)
 - User does not exist!
 - Wrong password!

2. Product Endpoints

These endpoints allow authenticated users to manage products in the inventory. All product endpoints require a valid JWT in the Authorization header.

Authentication Header Requirement:

For all product endpoints, you must include an Authorization header in the format:

Authorization: Bearer <your_access_token>

Replace <your_access_token> with the token obtained from the /login endpoint.

2.1. Add a New Product

- **URL:** /products
- **Method:** POST
- **Description:** Adds a new product record to the inventory. Requires authentication.
- **Request Body (JSON):**

```
{
```

```
"name": "Laptop Pro",
"type": "Electronics",
"sku": "LP-001-2024",
"quantity": 50,
"price": 1200.00,
"image_url": "https://example.com/laptop.jpg",
"description": "High performance laptop with 16GB RAM."
}
```

- **Responses:**

- **201 Created:**

```
{
  "message": "Product added successfully!",
  "product_id": "60d5ec49c6b6d8a4f8e7b9c0"
}
```

- **400 Bad Request:** If required product data (name, type, sku, quantity, price) is missing.

```
{
  "message": "Missing product data!"
}
```

- **401 Unauthorized:** If the Authorization header is missing or the token is invalid.

2.2. Get All Products

- **URL:** /products

- **Method:** GET

- **Description:** Retrieves a paginated list of all products in the inventory. Requires authentication.

- **Query Parameters:**

- **page** (optional, integer): The page number to retrieve. Default is 1.
 - **per_page** (optional, integer): The number of products per page. Default is 10.

- **Responses:**

- **200 OK:** A list of product objects.

```
[
  {
    "id": "60d5ec49c6b6d8a4f8e7b9c0",
    "name": "Laptop Pro",
```

```

    "type": "Electronics",
    "sku": "LP-001-2024",
    "image_url": "https://example.com/laptop.jpg",
    "description": "High performance laptop with 16GB RAM.",
    "quantity": 50,
    "price": 1200.00
  },
  {
    "id": "60d5ec49c6b6d8a4f8e7b9c1",
    "name": "Wireless Mouse",
    "type": "Accessories",
    "sku": "WM-002",
    "image_url": "",
    "description": "Ergonomic wireless mouse.",
    "quantity": 200,
    "price": 25.50
  }
]

```

- **400 Bad Request:** If page or per_page parameters are invalid.


```

{
  "message": "Invalid page or per_page parameter."
}

```
- **401 Unauthorized:** If the Authorization header is missing or the token is invalid.

2.3. Update Product Quantity

- **URL:** /products/<id>/quantity
- **Method:** PUT
- **Description:** Updates the quantity of a specific product identified by its ID. Requires authentication.
- **Path Parameters:**
 - id (string, required): The unique identifier of the product to update.
- **Request Body (JSON):**

```

{
  "quantity": 45
}

```

- **Responses:**

- **200 OK:**

```
{  
  "id": "60d5ec49c6b6d8a4f8e7b9c0",  
  "name": "Laptop Pro",  
  "quantity": 45,  
  "message": "Quantity updated successfully"  
}
```

- **400 Bad Request:** If the product ID format is invalid or the quantity is missing/not an integer.

```
{  
  "message": "Invalid product ID format!"  
}
```

or

```
{  
  "message": "Quantity is required and must be an integer!"  
}
```

- **404 Not Found:** If the product with the given ID does not exist.

```
{  
  "message": "Product not found!"  
}
```

- **401 Unauthorized:** If the Authorization header is missing or the token is invalid.

```
PS C:\Users\HP\Desktop\epify\backend> python test_api.py
--- Starting Inventory Management API Test Suite ---
Running: User Registration Test
✓ User Registration: PASSED
-----
Running: Login Test
✓ Login Test: PASSED
-----
Running: Add Product Test
✓ Add Product: PASSED
-----
Running: Update Quantity Test (New Quantity: 15)
✓ Update Quantity: PASSED
-----
Running: Get Products Test
✓ Get Products (Quantity Check): PASSED
-----

--- All tests completed successfully! ---
PS C:\Users\HP\Desktop\epify\backend>
```

TEST SCRIPT - PASSED

**YOU CAN ALSO SEE THE SWAGGER DOCUMENTATION OF THE ENDPOINTS AT
<http://localhost:8080/apidocs/>**