

1. API Documentation

a. API Base URL

- **Local:** `http://localhost:44313/api`

b. Authentication

All API endpoints (except authentication) require Bearer token:

```
Authorization: Bearer <access_token>
```

c. Endpoints

i. `/api/auth:`

- **POST `/login:`**
 - **Description:** Authenticate user and receive access token
 - **Headers:**
 - `Content-Type: application/json`
 - **Body:**

```
{
  "email": "admin@inventory.com",
  "password": "admin123"
}
```

- **Response:**

```
{
  "success": true,
  "message": "Login successful",
  "data": {
    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
    "user": {
      "userId": 123,
      "email": "admin@inventory.com",
      "fullName": "Admin",
      "role": "System Administrator"
    },
    "expiresAt": "2025-06-22T10:30:00.000Z"
  }
}
```

- **POST /register:**

- **Description:** Register new user account
- **Body:**

```
{
  "email": "newuser@example.com",
  "password": "securepassword123",
  "fullName": "Example Name",
  "confirmPassword": "securepassword123"
}
```

- **POST /forgot-password:**

- **Description:** Initiate password reset with OTP
- **Body:**

```
{
  "email": "user@example.com"
}
```

ii. /api/products:

- **GET /:**

- **Description:** Retrieve all products with optional filtering
- **Headers:**
 - Authorization: Bearer <access_token>

- **Query Params:**

- `categoryId` (int, optional): Filter by category
- `search` (string, optional): Search in name/description
- `lowStock` (boolean, optional): Filter low stock items
- `page` (int, optional): Page number (default: 1)
- `pageSize` (int, optional): Items per page (default: 50)

- **Response:**

```
{
  "success": true,
  "data": {
    "products": [
      {
        "productId": 1,
        "name": "Laptop Computer",
        "description": "High-performance laptop",
        "price": 1299.99,
        "minimumStockLevel": 5,
        "categoryId": 2,
        "categoryName": "Electronics",
        "currentStock": 15,
        "isLowStock": false
      }
    ],
    "pagination": {
      "currentPage": 1,
      "totalPages": 5,
      "totalItems": 247
    }
  }
}
```

- **GET /:**

- **Description:** Retrieve specific product by ID

- **Headers:**

- `Authorization: Bearer <access_token>`

- **Response:**

```
{
  "success": true,
  "data": {
    "productId": 1,
    "name": "Laptop Computer",
    "description": "High-performance laptop",
    "price": 1299.99,
    "currentStock": 15,
    "categoryName": "Electronics"
  }
}
```

- **POST /:**

- **Description:** Create new product (Admin/Manager only)
- **Body:**

```
{
  "name": "Wireless Mouse",
  "description": "Ergonomic wireless mouse",
  "price": 29.99,
  "minimumStockLevel": 20,
  "categoryId": 2
}
```

- **PUT /:**

- **Description:** Update existing product (Admin/Manager only)

- **DELETE /:**

- **Description:** Delete product (Admin only)

iii. /api/categories:

- **GET /:**

- **Description:** Retrieve all categories

- **POST /:**

- **Description:** Create new category (Admin/Manager only)

iv. /api/inventory:

- **GET /:**

- **Description:** Retrieve current inventory levels

- **PUT /:**

- **Description:** Update inventory quantity

v. /api/orders:

- **GET /:**
 - **Description:** Retrieve all orders
- **POST /:**
 - **Description:** Create new purchase order

vi. /api/notifications:

- **GET /:**
 - **Description:** Retrieve user notifications
- **PUT //read:**
 - **Description:** Mark notification as read

d. General Request & Response Conventions

Standard Response Format:

```
{
  "success": true/false,
  "message": "Operation description",
  "data": { /* Response payload */ },
  "errorType": "ErrorType" (if error),
  "timestamp": "2025-01-20T10:30:00.000Z"
}
```

HTTP Status Codes:

- 200: Success (GET, PUT, PATCH)
 - 201: Created (POST)
 - 204: No Content (DELETE)
 - 400: Bad Request
 - 401: Unauthorized
 - 403: Forbidden
 - 404: Not Found
 - 422: Validation Error
 - 500: Internal Server Error
-