Inventory Management API - JSON Payloads

Product Management Endpoints

POST /api/products - Create New Product

Required Fields:

- `name` (string) - Product name
- `sku` (string) - Unique product SKU

Optional Fields:

- `description` (string) - Product description
- `stock_level` (integer) - Initial stock quantity (default: 0)
- `min_stock_threshold` (integer) - Low stock alert threshold (default: 10)
- `price` (float) - Product price (default: 0.0)

```json
{
  "name": "Wireless Bluetooth Headphones",
  "sku": "WBH-001",
  "description": "Premium wireless headphones with noise cancellation",
  "stock_level": 50,
  "min_stock_threshold": 15,
  "price": 99.99
}
```

Minimal Payload:

```json
{
  "name": "Basic Product",
  "sku": "BP-001"
}
```

PUT /api/products/<id> - Update Product

All fields are optional - only include fields you want to update:

```json
{
  "name": "Updated Product Name",
  "description": "Updated description",
  "stock_level": 75,
  "min_stock_threshold": 20,
  "price": 129.99
}
```

Update only stock level:

```json
{
  "stock_level": 100
}
```

Update only threshold:

```json
{
  "min_stock_threshold": 25
}
```

Restocking Operations

POST /api/products/<id>/restock - Restock Product

Required Fields:

- `quantity` (integer) - Number of units to add (must be positive)

Optional Fields:

- `notes` (string) - Restocking notes/comments

```json
{
  "quantity": 25,
  "notes": "Weekly inventory replenishment from supplier ABC"
}
```

Minimal Payload:

```json
{
  "quantity": 10
}
```

Response Examples

Successful Product Creation Response

```json
{
  "success": true,
  "message": "Product created successfully",
  "product": {
    "id": 1,
    "name": "Wireless Bluetooth Headphones",
    "sku": "WBH-001",
    "description": "Premium wireless headphones with noise cancellation",
    "stock_level": 50,
    "min_stock_threshold": 15,
    "price": 99.99,
    "created_at": "2025-06-23T10:30:00",
    "updated_at": "2025-06-23T10:30:00",
    "is_low_stock": false
  }
}
```

Successful Restock Response

```json
{
  "success": true,
  "message": "Product restocked successfully. Added 25 units.",
  "product": {
    "id": 1,
    "name": "Wireless Bluetooth Headphones",
    "sku": "WBH-001",
    "description": "Premium wireless headphones with noise cancellation",
    "stock_level": 75,
    "min_stock_threshold": 15,
    "price": 99.99,
    "created_at": "2025-06-23T10:30:00",
    "updated_at": "2025-06-23T11:45:00",
    "is_low_stock": false
  },
  "restock_log": {
    "id": 1,
    "product_id": 1,
    "product_name": "Wireless Bluetooth Headphones",
    "product_sku": "WBH-001",
    "quantity_added": 25,
    "previous_stock": 50,
    "new_stock": 75,
    "restocked_at": "2025-06-23T11:45:00",
    "notes": "Weekly inventory replenishment from supplier ABC"
  }
}
```

Error Response Examples

```json
{
  "success": false,
  "error": "Name and SKU are required"
}
```

```json
{
  "success": false,
  "error": "Product with this SKU already exists"
}
```

```json
{
  "success": false,
  "error": "Quantity must be positive"
}
```

GET Endpoints (No Payloads Required)

These endpoints don't require JSON payloads, only URL parameters:

- `GET /api/products` - No payload

- `GET /api/products/<id>` - No payload

- `GET /api/restocks?page=1&per_page=50` - Optional query parameters

- `GET /api/products/low-stock` - No payload

- `GET /api/products/analytics` - No payload

- `DELETE /api/products/<id>` - No payload

Common Validation Rules

1. **SKU must be unique** - Cannot create products with duplicate SKUs

2. **Quantity must be positive** - Restocking quantity must be > 0

3. **Required fields** - Name and SKU are mandatory for product creation

4. **Data types** - Ensure correct types (integers for quantities, floats for prices)

5. **Stock levels** - Cannot be negative

6. **Price** - Should be non-negative float value

Example Usage Scenarios

Creating a new product with full details:

```json
{
  "name": "Gaming Mouse",
  "sku": "GM-2024-001",
  "description": "High-precision gaming mouse with RGB lighting",
  "stock_level": 30,
  "min_stock_threshold": 5,
  "price": 79.99
}
```

Emergency restock with notes:

```json
{
  "quantity": 100,
  "notes": "Emergency restock due to unexpected high demand"
}
```

Updating only the low stock threshold:

```json
{
  "min_stock_threshold": 8
}
```