

Simple Inventory System

API Documentation:

Register a new user

POST /api/users/register
Content-Type: application/json

Body:

```
{  
  "email": "test@example.com",  
  "password": "test1234",  
  "name": "Test User"  
}
```

Return:

```
{  
  "message": "User registered successfully"  
}
```

Or

```
{  
  "message": "User Already Exists"  
}
```

Login with a user

POST /api/users/login
Content-Type: application/json

Body:

```
{  
  "email": "test@example.com",  
  "password": "test1234"  
}
```

Return:

```
{  
  "token": "<jwt-token>"  
}
```

Or

```
{
  "message": "User not found"
}
```

Create a new product

POST /api/products
Content-Type: application/json
Authorization: Bearer <jwt-token>

Body:

```
{
  "name": "Product 1",
  "price": 10.99,
  "quantity": 100
}
```

Return:

```
{
  "message": "Product created successfully"
}
```

Or

```
{
  "message": "Invalid Token"
}
```

Get all products

GET /api/products/

Return :

Array Of Objects

```
[
  {
    "_id": "64563d1100690d3471b4e56a",
    "name": "Product 2",
    "price": 10000,
```

```
    "quantity": 1,  
    "userId": "64563c7e00690d3471b4e55c",  
    "__v": 0  
  }  
]
```

Get a specific product

GET /api/products/<product-id>

Return:

```
{  
  "_id": "64563d1100690d3471b4e56a",  
  "name": "Product 2",  
  "price": 10000,  
  "quantity": 1,  
  "userId": "64563c7e00690d3471b4e55c",  
  "__v": 0  
}
```

Or

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

Update a product

PUT /api/products/<product-id>

Content-Type: application/json

Authorization: Bearer <jwt-token>

Body:

```
{  
  "name": "Product 1",  
  "price": 12.99  
}
```

Return:

```
{  
  "message": "Product updated successfully"  
}
```

Or

```
{  
  "message": "Invalid Token"  
}
```

Delete a product

DELETE /api/products/<product-id>

Authorization: Bearer <jwt-token>

Return:

```
{  
  "message": "Product deleted successfully"  
}
```

Or

```
{  
  "message": "Invalid Token"  
}
```