Prepared by Temirbay Assem SE-2301

Clothing Store DB

Advanced Databases (NoSQL)

Shynar Akhmetzhanova

GitHub

Introduction

Aim: To develop a backend system for an e-commerce platform that efficiently handles product management, order processing, and user authentication.

Goals:

- Implement a **scalable** and **secure** backend architecture.
- Provide **CRUD operations** for products, users, and orders.
- Ensure **fast query performance** using indexing and optimization.
- Implement **authentication** & **authorization** for different user roles.

Relevance

Why is this project important?

- The e-commerce market is growing rapidly, and businesses require efficient backend solutions to manage large inventories and customer transactions.
- I analyzed the backend structures of Zara, H&M, Shein, and ASOS.
    - Zara & ASOS have similar website structures and data storage models.
    - ASOS is more user-friendly and provides a better product filtering experience.
- My project aims to optimize product search, order handling, and user experience by learning from these platforms.

Research – Existing E-Commerce Platforms

Zara:

Stylish UI, strong branding
Limited filtering options, slower backend

H&M:

Good product categorization
Occasional slow search functionality

Shein

Large inventory, affordable pricing
Lower quality control, inconsistent backend

Shein

User-friendly search, fast backend
Product return process can be improved

ASOS offers the best balance of speed, structure, and usability, which I used as a reference for our backend.

System Architecture

My backend system follows a RESTful API architecture and includes:

Authentication & Authorization

- Using JWT for secure user access

Database Management

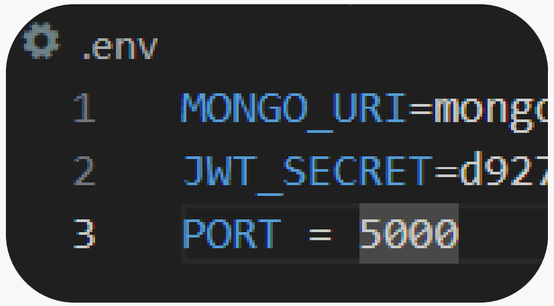
- MongoDB for handling large datasets efficiently

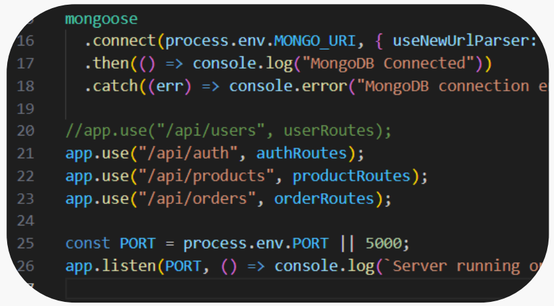
API Design

- Endpoints for managing users, products, and orders

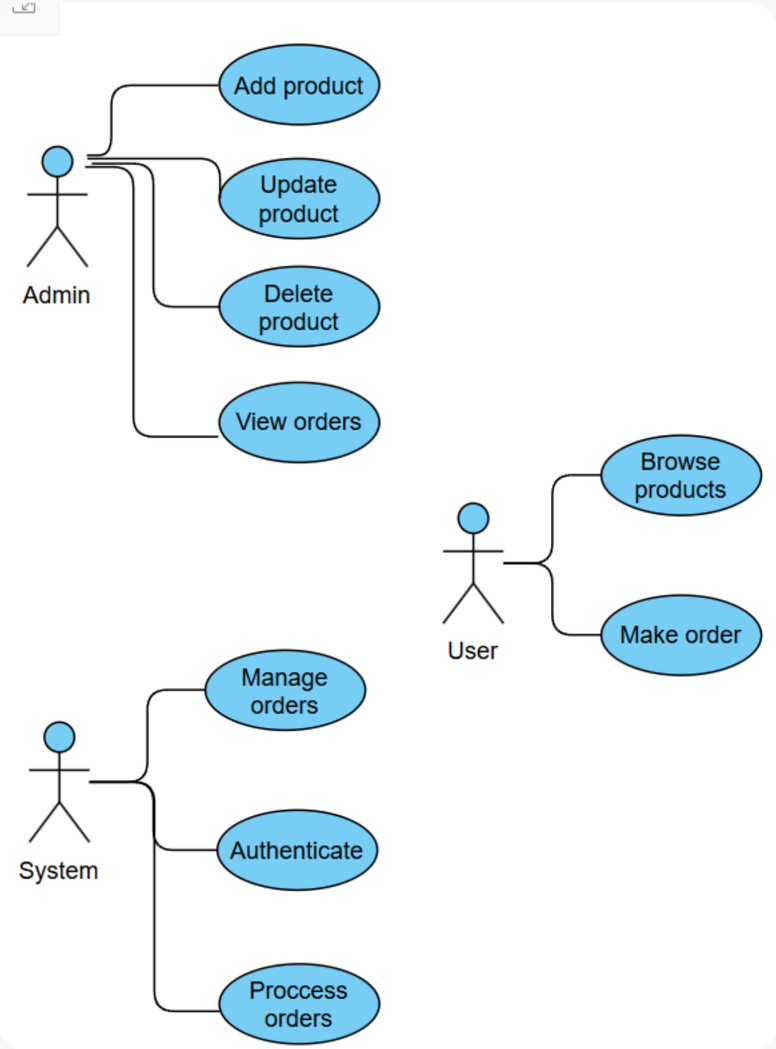
Security Measures

- User role-based access (admin vs. customer)





Diagrams



UML Diagrams – Use-Case Diagram:

- Shows interactions between users (customers, admins) and the system.
- Customers can register, log in, browse products, and place orders.
- Admins can manage products and orders.

Diagrams

Database Design – ERD Diagram:

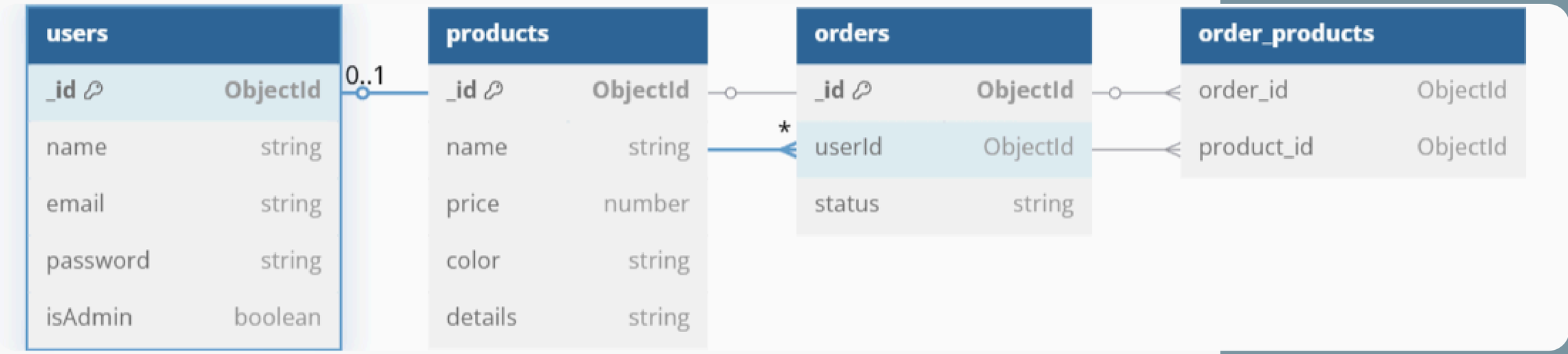
Three main collections: Users, Products, Orders.

Relationships:

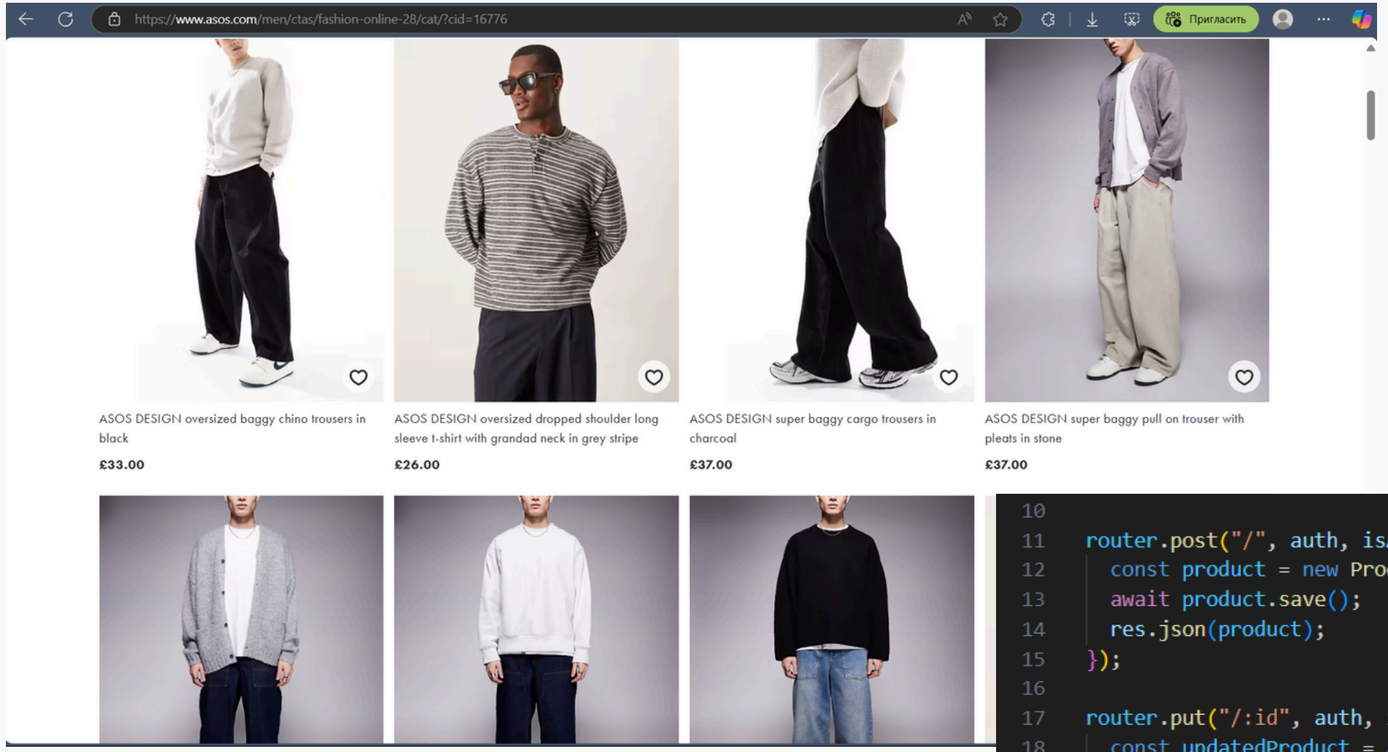
- Users → Orders (One-to-Many)
- Products → Orders (Many-to-Many)

Indexes:

- ProductSchema.index({ name: 1 }) for fast product search.
- OrderSchema.index({ userId: 1 }) for efficient order history retrieval.



Data Collection



How did I obtain the data?

1. Web Parsing: I parsed product data from ASOS.
2. APIs: Used external APIs for currency conversion.
3. Manual Entry: Admins can manually add products

```
10
11  router.post("/", auth, isAdmin, async (req, res) => {
12    const product = new Product(req.body);
13    await product.save();
14    res.json(product);
15  });
16
17  router.put("/:id", auth, isAdmin, async (req, res) => {
18    const updatedProduct = await Product.findByIdAndUpdate(req.params.id, req.body, { new: true });
19    res.json(updatedProduct);
20  });
21
22  router.delete("/:id", auth, isAdmin, async (req, res) => {
23    await Product.findByIdAndDelete(req.params.id);
24    res.json({ message: "Товар удален" });
25  });
```

Features

✅ CRUD Operations:

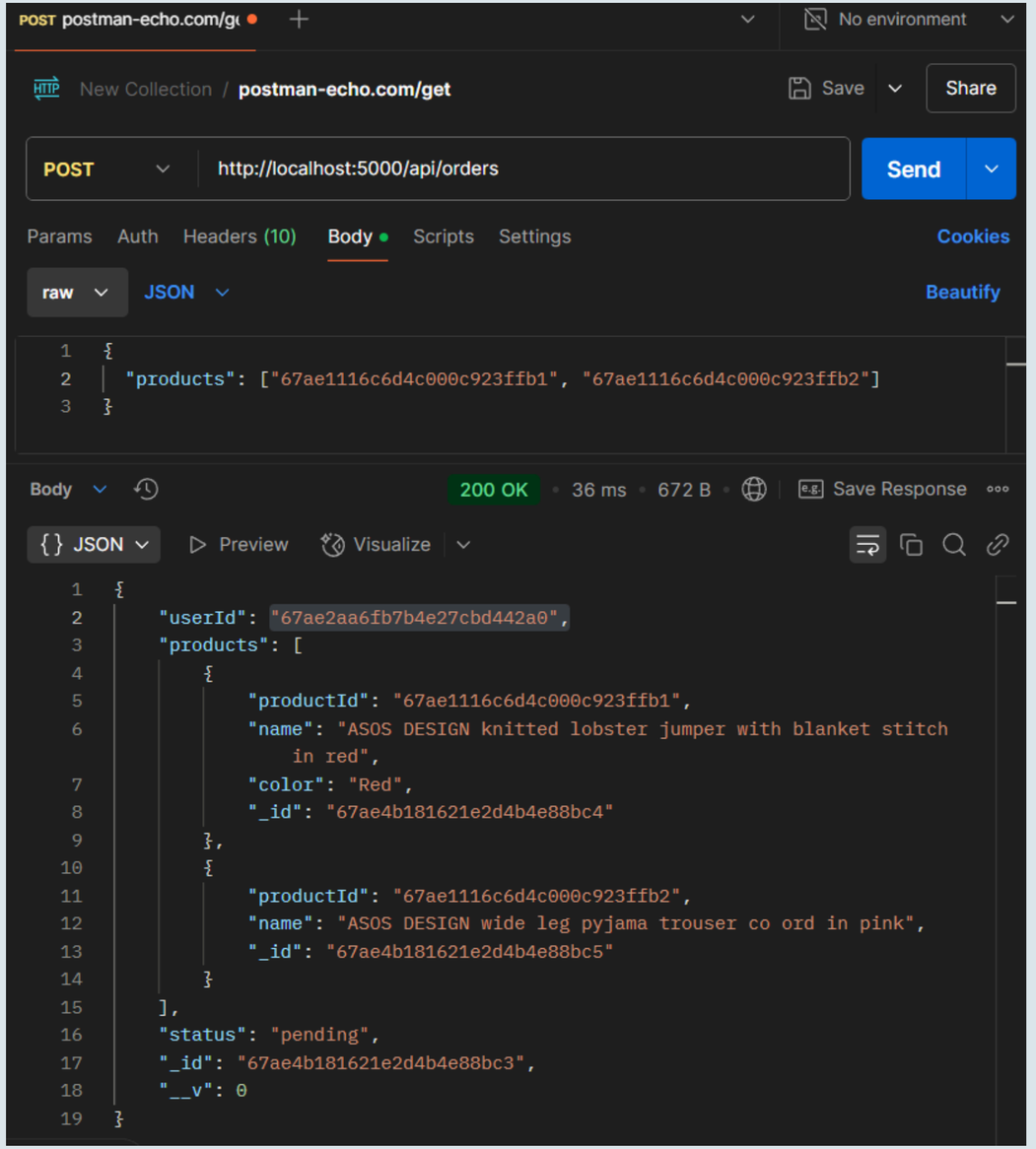
- Products: Add, update, delete, and view product details.
- Users: Register, authenticate, and manage profiles.
- Orders: Place orders, track status, and manage past purchases.

✅ Query Optimization:

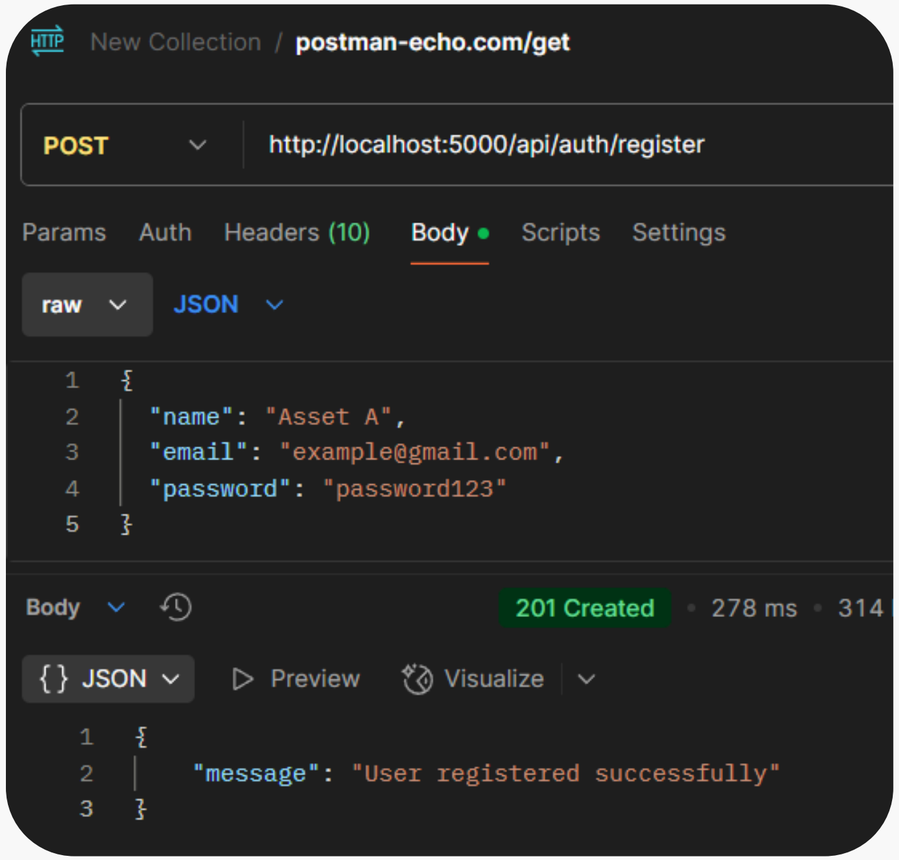
- Used MongoDB Indexing to speed up search queries.

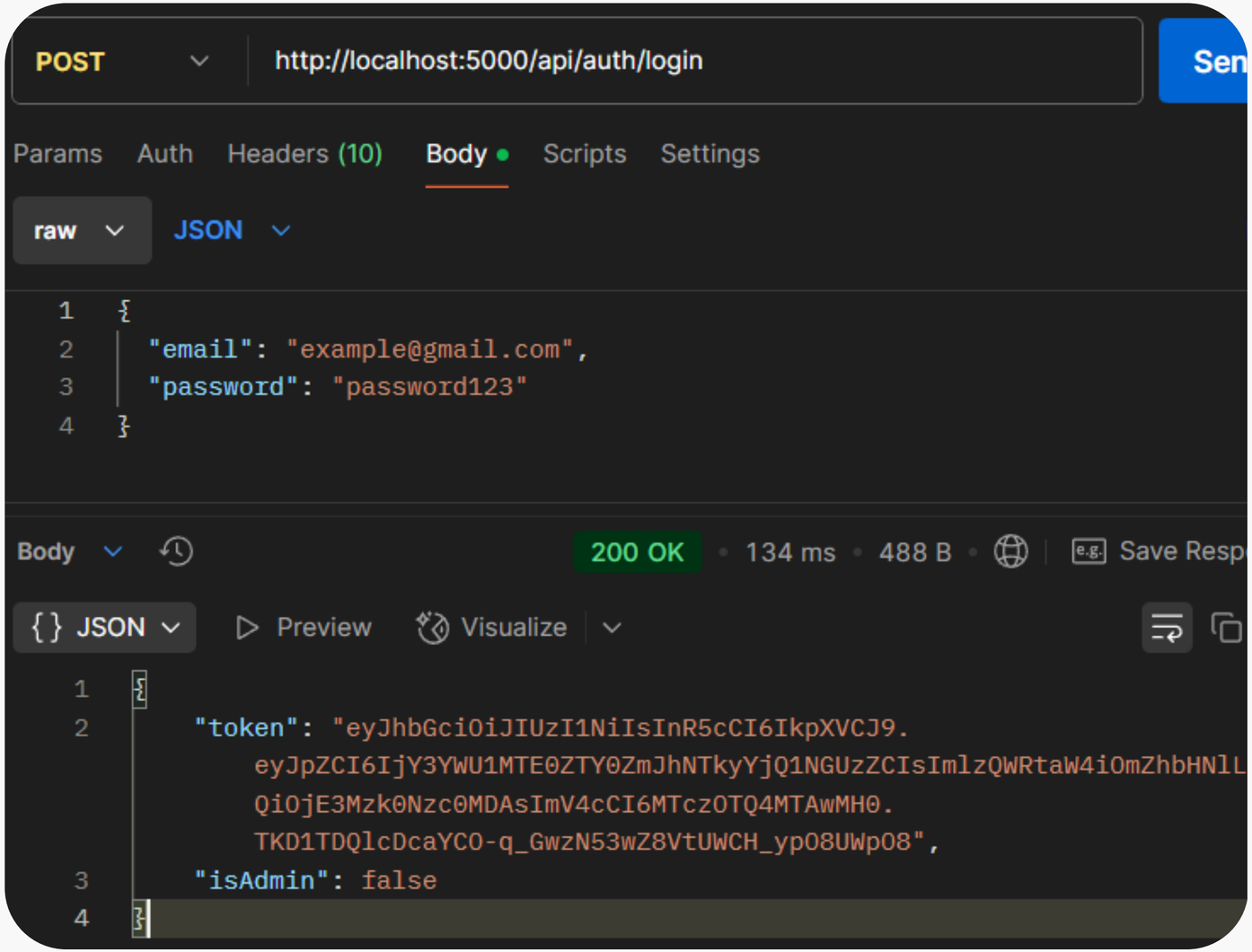
✅ Security Aspects:

- JWT Authentication: Secure login system.
- Role-based Access: Admins manage products/orders; users only place orders.

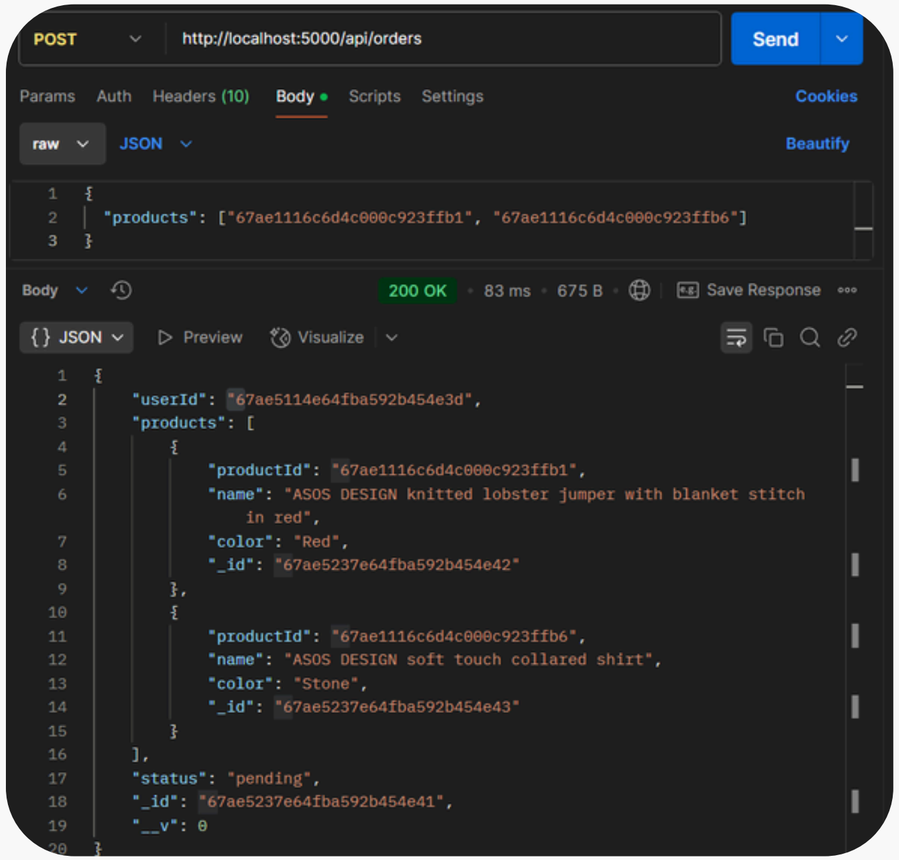


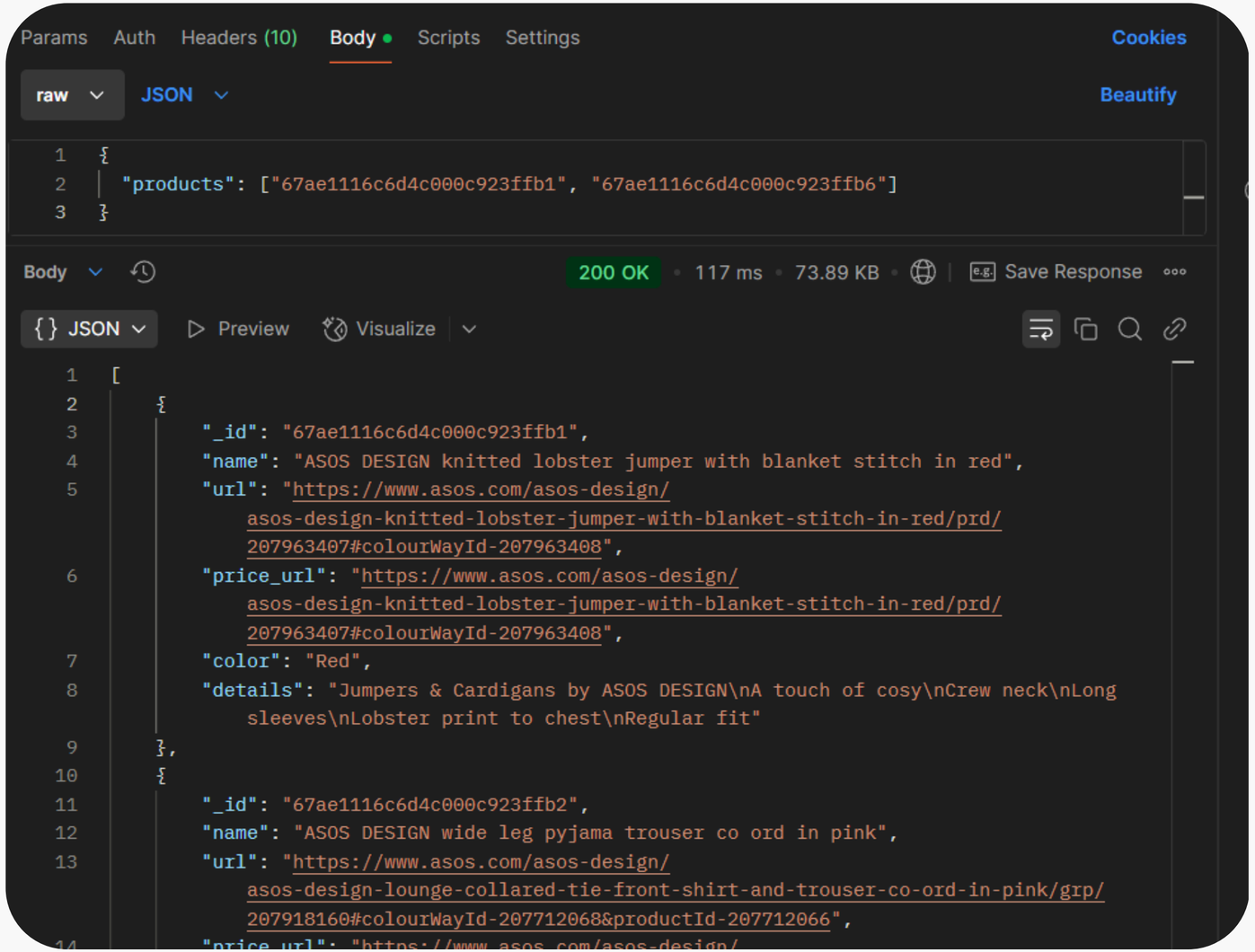
Demo


New Collection / postman-echo.com/get
POST
http://localhost:5000/api/auth/register
Params Auth Headers (10) Body Scripts Settings
raw JSON
{
"name": "Asset A",
"email": "example@gmail.com",
"password": "password123"
}
Body
201 Created
278 ms
JSON Preview Visualize
{
"message": "User registered successfully"
}


POST
http://localhost:5000/api/auth/login
Sen
Params Auth Headers (10) Body Scripts Settings
raw JSON
{
"email": "example@gmail.com",
"password": "password123"
}
Body
200 OK
134 ms
488 B
Save Resp
JSON Preview Visualize
{
"token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
eyJpZCI6IjY3YWU1MTE0ZTY0ZmJhNTkyYjQ1NGUzZCIsImlzQWRtaW4iOmZhbHNlL
QiOjE3Mzk0Nzc0MDAsImV4cCI6MTczOTQ4MTAwMH0.
TKD1TDQlcDcaYCO-q_GwzN53wZ8VtUWCH_ypO8UWpO8",
"isAdmin": false
}

Demo


POST
http://localhost:5000/api/orders
Send
Params Auth Headers (10) Body Scripts Settings
Cookies
raw JSON
Beautify
{
"products": ["67ae1116c6d4c000c923ffb1", "67ae1116c6d4c000c923ffb6"]
}
Body
200 OK 83 ms 675 B
Save Response
{ } JSON
Preview
Visualize
{
"userId": "67ae5114e64fba592b454e3d",
"products": [
{
"productId": "67ae1116c6d4c000c923ffb1",
"name": "ASOS DESIGN knitted lobster jumper with blanket stitch in red",
"color": "Red",
"_id": "67ae5237e64fba592b454e42"
},
{
"productId": "67ae1116c6d4c000c923ffb6",
"name": "ASOS DESIGN soft touch collared shirt",
"color": "Stone",
"_id": "67ae5237e64fba592b454e43"
}
],
"status": "pending",
"_id": "67ae5237e64fba592b454e41",
"__v": 0
}


Params Auth Headers (10) Body Scripts Settings
Cookies
raw JSON
Beautify
{
"products": ["67ae1116c6d4c000c923ffb1", "67ae1116c6d4c000c923ffb6"]
}
Body
200 OK 117 ms 73.89 KB
Save Response
{ } JSON
Preview
Visualize
[
{
"_id": "67ae1116c6d4c000c923ffb1",
"name": "ASOS DESIGN knitted lobster jumper with blanket stitch in red",
"url": "https://www.asos.com/asos-design/asos-design-knitted-lobster-jumper-with-blanket-stitch-in-red/prd/207963407#colourWayId-207963408",
"price_url": "https://www.asos.com/asos-design/asos-design-knitted-lobster-jumper-with-blanket-stitch-in-red/prd/207963407#colourWayId-207963408",
"color": "Red",
"details": "Jumpers & Cardigans by ASOS DESIGN\nA touch of cosy\nCrew neck\nLong sleeves\nLobster print to chest\nRegular fit"
},
{
"_id": "67ae1116c6d4c000c923ffb2",
"name": "ASOS DESIGN wide leg pyjama trouser co ord in pink",
"url": "https://www.asos.com/asos-design/asos-design-lounge-collared-tie-front-shirt-and-trouser-co-ord-in-pink/grp/207918160#colourWayId-207712068&productId-207712066",

Conclusion

Key Findings:

- Backend inspired by ASOS due to its efficient architecture.
- Implemented secure authentication and optimized database queries.
- MongoDB indexes improved search speed and order retrieval.

🔮 **Future Improvements:**

- Implement a recommendation system based on user purchases.
- Add payment gateway integration for real transactions.
- Improve admin dashboard for better order management

Thank you