

BÁO CÁO THỐNG KÊ VÀ CẢI TIẾN

1. Báo cáo kết quả kiểm thử

1.1. Kiểm thử Đơn vị (Unit Test)

- Công cụ: Jest
- Số lượng: 5 Test Case
- Thông qua: 5 (100%)
- Hình ảnh:

```
PS C:\Users\huynh\OneDrive\Desktop\Programming\WEBSITE CNPM\Backend> npm test

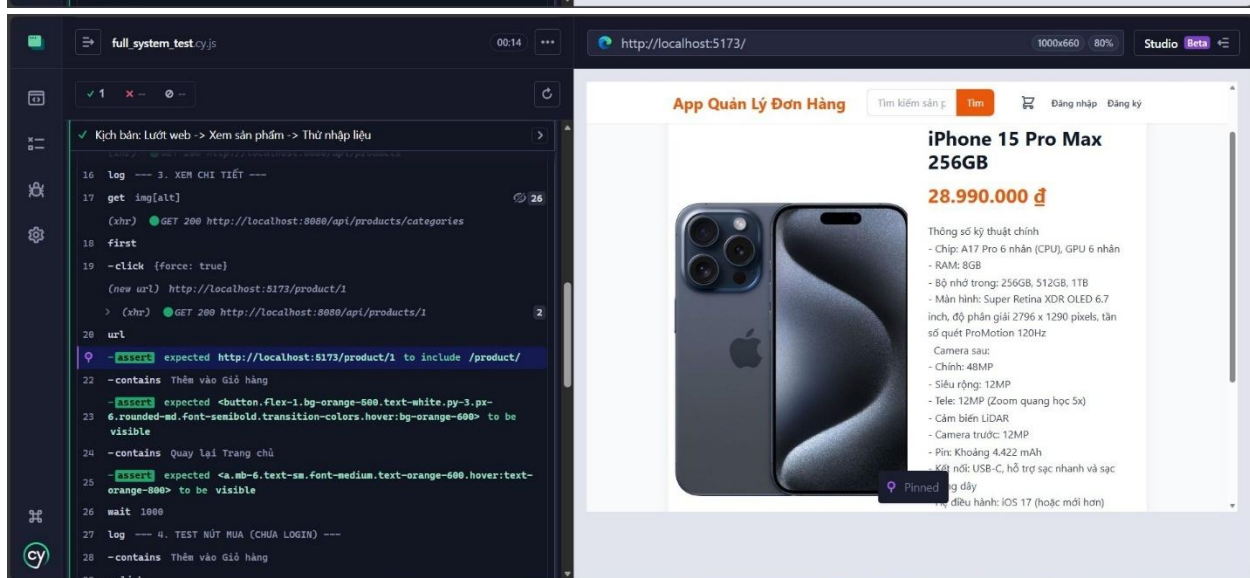
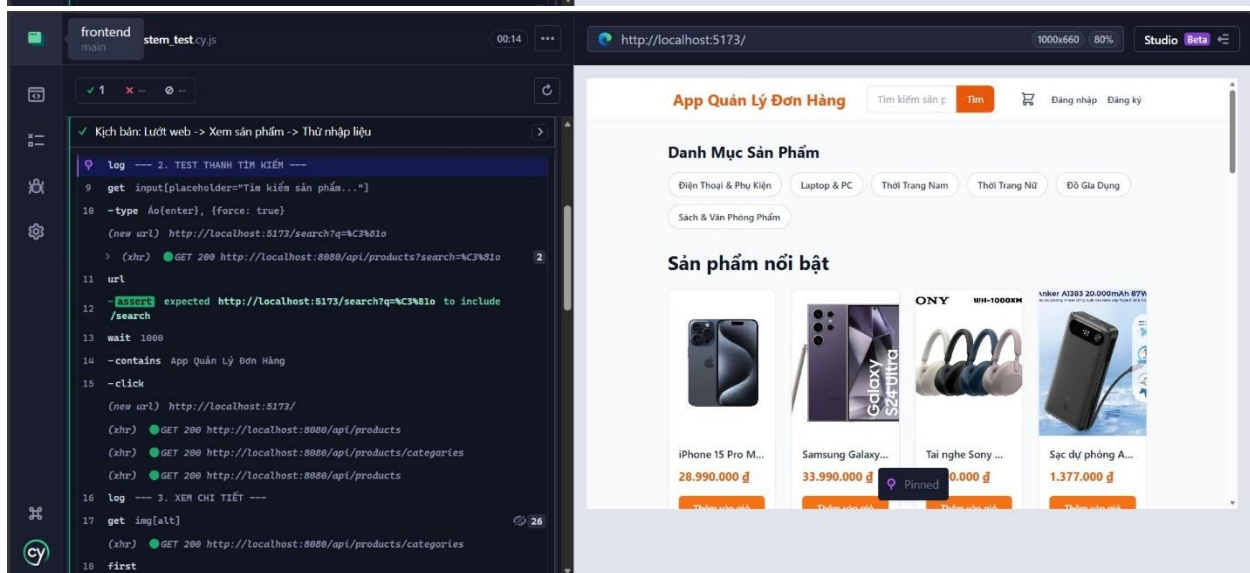
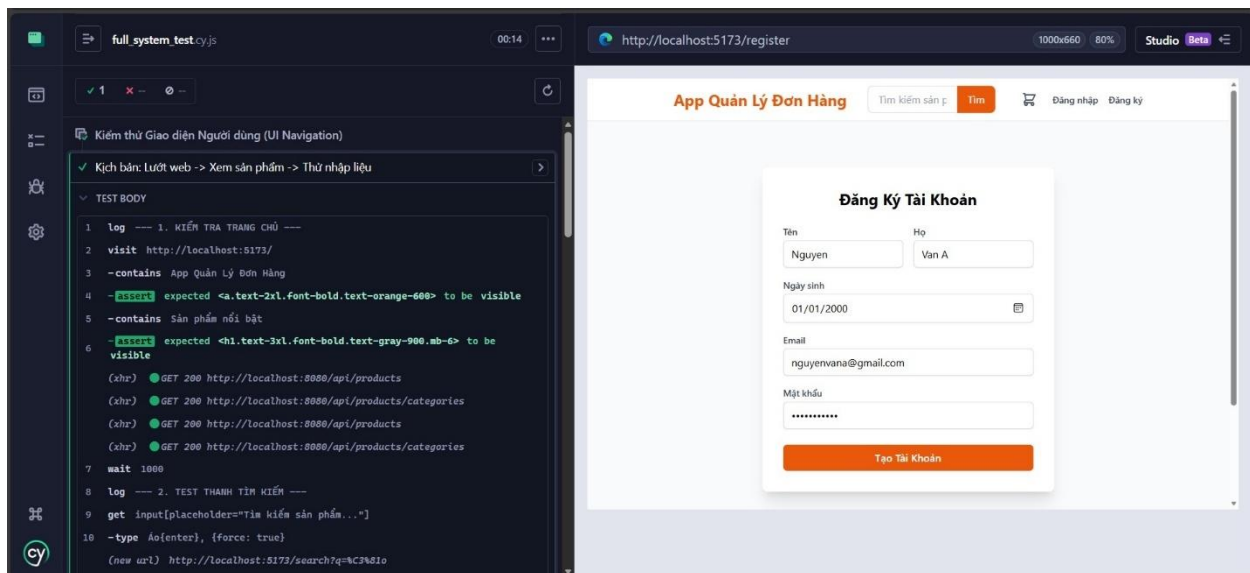
> backend@1.0.0 test
> jest

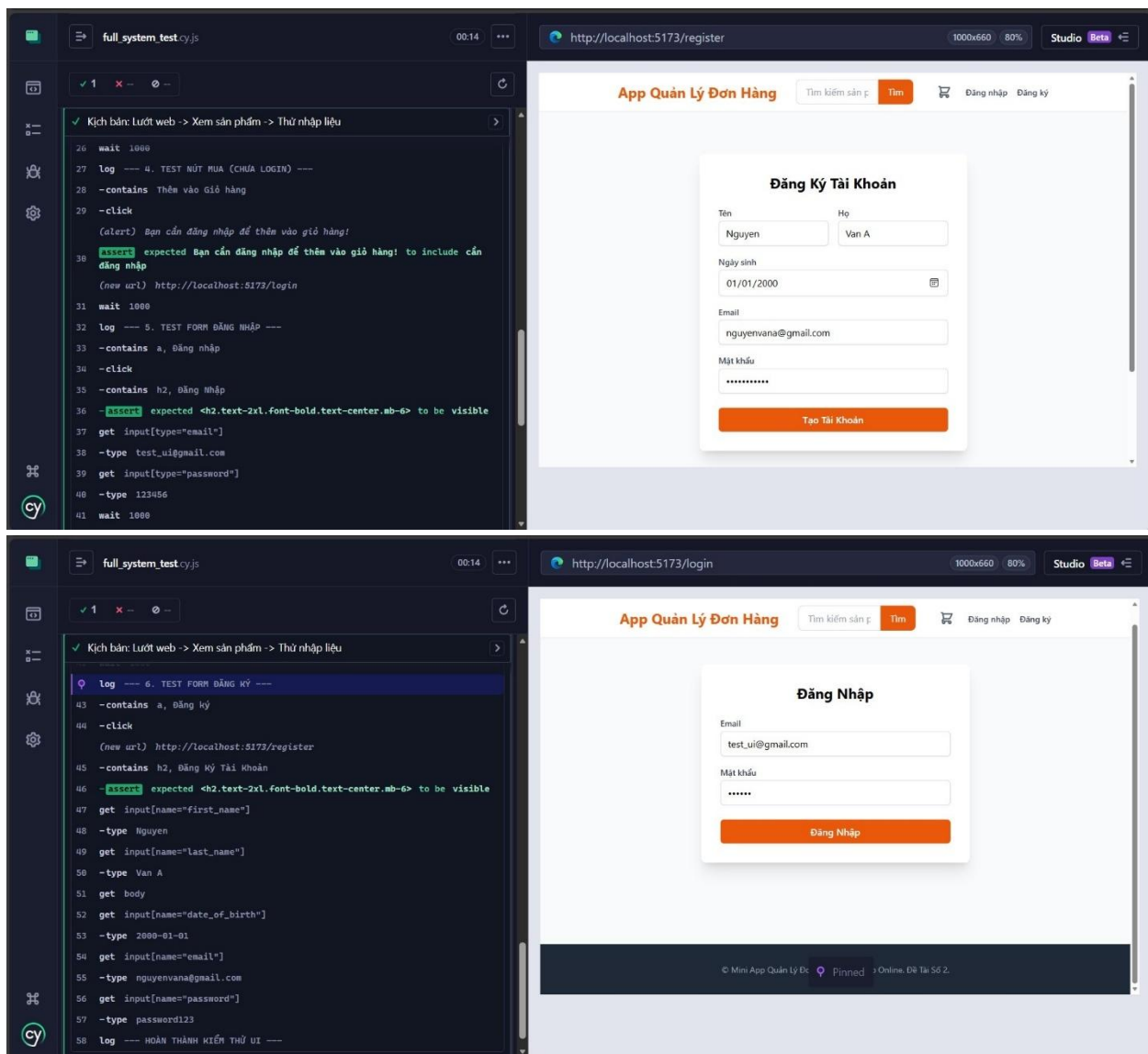
PASS tests/orderUtils.test.js
  Unit Test: Hàm calculateTotal (Tính tổng tiền)
    ✓ TC-01: Tính đúng tổng tiền cho giỏ hàng có 2 món (3 ms)
    ✓ TC-02: Trả về 0 nếu giỏ hàng rỗng (1 ms)
    ✓ TC-03: Báo lỗi nếu dữ liệu sai (Giá âm) (8 ms)
  Unit Test: Hàm validateOrderInput (Kiểm tra đầu vào)
    ✓ TC-04: Báo lỗi nếu thiếu địa chỉ giao hàng (1 ms)
    ✓ TC-05: Hợp lệ nếu đủ thông tin và đúng phương thức (1 ms)

Test Suites: 1 passed, 1 total
Tests:       5 passed, 5 total
Snapshots:   0 total
Time:        0.57 s, estimated 1 s
Ran all test suites.
```

1.2. Kiểm thử Giao diện & Tích hợp (UI/Integration Test)

- Công cụ: Cypress
- Hình ảnh:





1.3. Kiểm thử API (API Test)

- **Phương thức:** Thunder Client
- **Số lượng:** 4 Collections (Auth, Products, Cart, Orders)
- **Thông qua:** 4 (100%)

❖ Auth (Quản lý khách hàng)

1. Đăng ký (POST /register)

POST http://localhost:8080/api/auth/register Send

Query Headers Auth **Body** Tests Pre Run

JSON XML Text Form Form-encode GraphQL Binary

JSON Content Format

```
1 {
2   "first_name": "A",
3   "last_name": "Nguyễn Văn",
4   "email": "customer123@gmail.com",
5   "password": "password123",
6   "date_of_birth": "2005-01-01"
7 }
```

Status: 201 Created Size: 222 Bytes Time: 174 ms

Response Headers Cookies Results Docs {} ≡

```
1 {
2   "message": "Đăng ký thành công!",
3   "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6NCwicm9sZSI6ImN1c3RvbWV5IiwiaWF0IjoxNzYzNmMxMDQ3LCJleHAiOjE3NjMzMzQ2NDd9.Le6pRbGgOfU7Z8_I-dB5y1-2MjAcvm0zLF8XqBkk3Q8",
4   "user_id": 4
5 }
```

Trường hợp trùng email thì sẽ không thể đăng ký được

POST http://localhost:8080/api/auth/register Send

Query Headers Auth **Body** Tests Pre Run

JSON XML Text Form Form-encode GraphQL Binary

JSON Content Format

```
1 {
2   "first_name": "ABC",
3   "last_name": "Nguyễn Văn",
4   "email": "customer123@gmail.com",
5   "password": "password123",
6   "date_of_birth": "2005-01-01"
7 }
```

Status: 400 Bad Request Size: 51 Bytes Time: 5 ms

Response Headers Cookies Results Docs {} ≡

```
1 {
2   "message": "Email này đã được sử dụng."
3 }
```

2. Đăng nhập (POST /login)

POST http://localhost:8080/api/auth/login Send

Query Headers Auth **Body** Tests Pre Run

JSON XML Text Form Form-encode GraphQL Binary

JSON Content Format

```
1 {
2   "email": "customer@gmail.com",
3   "password": "password123"
4 }
5
```

Status: 200 OK Size: 243 Bytes Time: 138 ms

Response Headers Cookies Results Docs {} ≡

```
1 {
2   "message": "Đăng nhập thành công!",
3   "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6Mywicm9sZSI6ImN1c3RvbWV5IiwiaWF0IjoxNzYzNmY0MjA2LCJleHAiOjE3NjMzMjc4MDZ9.iKEHwtOUTcbV9wqPgpr1sPN94Fiqkv2muqJm8MF1KBg",
4   "user_id": 3,
5   "role": "customer"
6 }
```

Trường hợp đăng nhập nhưng sai mật khẩu hoặc email

The screenshot shows a REST client interface with a POST request to `http://localhost:8080/api/auth/login`. The request body is a JSON object: `{ "email": "customer@gmail.com", "password": "password12345" }`. The response status is `401 Unauthorized` with a size of 44 bytes and a time of 130 ms. The response body is a JSON object: `{ "message": "Sai email hoặc mật khẩu." }`.

3. Cập nhật thông tin người dùng (PUT /profile)

Chỉ có thể cập nhật thông tin sau khi người dùng đăng nhập

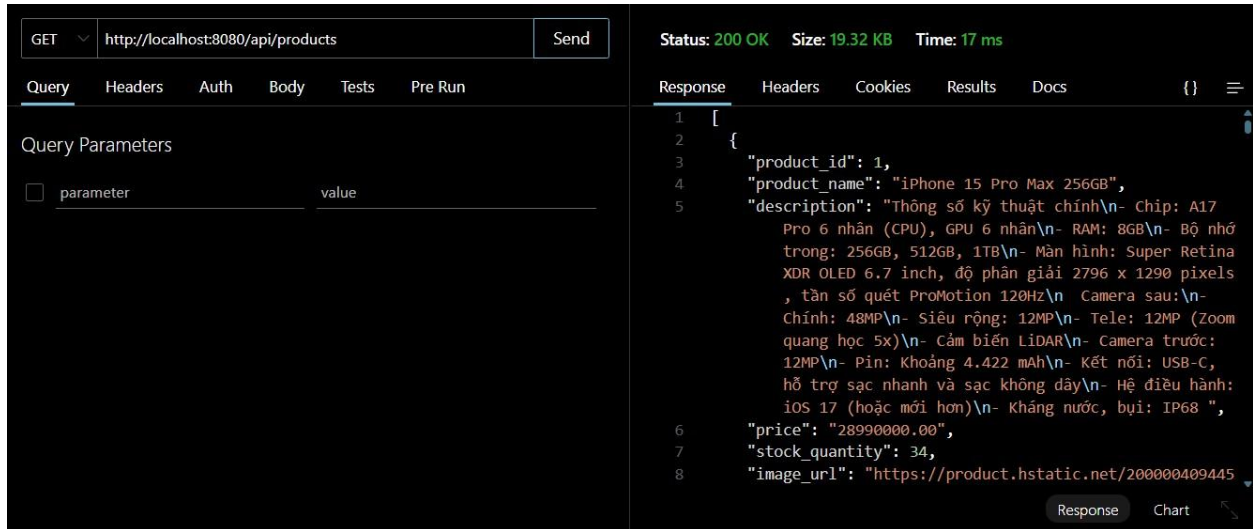
The screenshot shows a REST client interface with a PUT request to `http://localhost:8080/api/auth/profile`. The request body is a JSON object: `{ "first_name": "D", "last_name": "Nguyễn Văn", "shipping_address": "123 Đường A, TPHCM", "phone_number": "0123456789" }`. The response status is `200 OK` with a size of 283 bytes and a time of 34 ms. The response body is a JSON object: `{ "message": "Cập nhật thông tin thành công!", "user": { "customer_id": 3, "email": "customer@gmail.com", "first_name": "D", "last_name": "Nguyễn Văn", "shipping_address": "123 Đường A, TPHCM", "phone_number": "0123456789", "role": "customer", "date_of_birth": "2004-12-31T17:00:00.000Z" } }`.

Trường hợp người dùng chưa đăng nhập

The screenshot shows a REST client interface with a PUT request to `http://localhost:8080/api/auth/profile`. The request body is a JSON object: `{ "first_name": "ABC", "last_name": "Nguyễn Văn", "shipping_address": "123 Đường ADE, TPHCM", "phone_number": "0123456789" }`. The response status is `401 Unauthorized` with a size of 52 bytes and a time of 6 ms. The response body is a JSON object: `{ "message": "Chưa đăng nhập, không có token." }`.

❖ Products (Quản lý sản phẩm)

1. Lấy danh sách sản phẩm (GET /products)



GET Send

Query Headers Auth Body Tests Pre Run

Query Parameters

<input type="checkbox"/>	parameter	value
--------------------------	-----------	-------

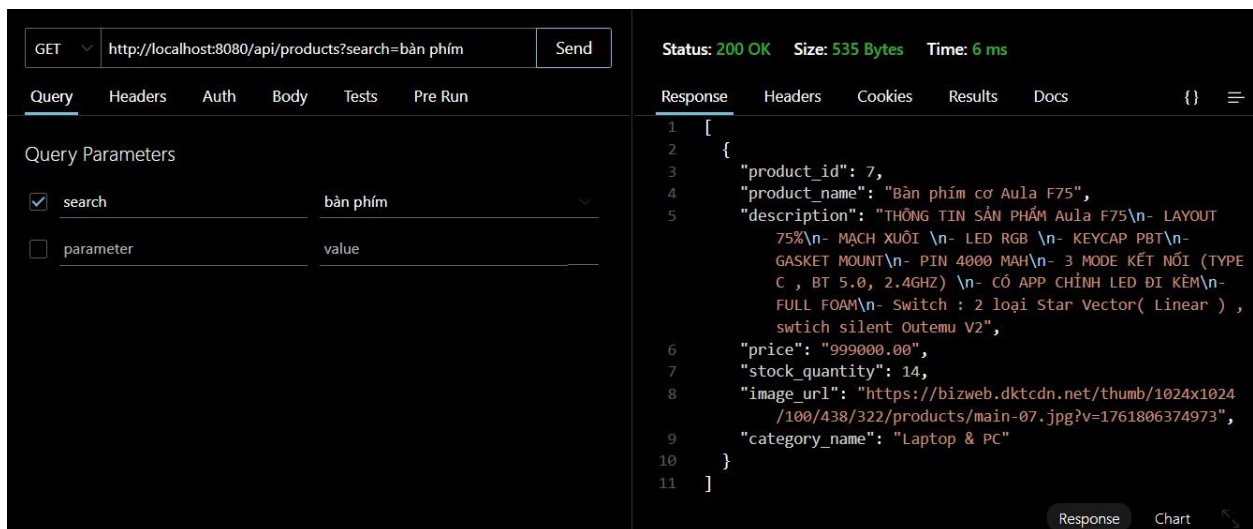
Status: 200 OK Size: 19.32 KB Time: 17 ms

Response Headers Cookies Results Docs {} ≡

```
1 [
2   {
3     "product_id": 1,
4     "product_name": "iPhone 15 Pro Max 256GB",
5     "description": "Thông số kỹ thuật chính\n- Chip: A17
Pro 6 nhân (CPU), GPU 6 nhân\n- RAM: 8GB\n- Bộ nhớ
trong: 256GB, 512GB, 1TB\n- Màn hình: Super Retina
XDR OLED 6.7 inch, độ phân giải 2796 x 1290 pixels
, tần số quét ProMotion 120Hz\n- Camera sau:\n-
Chính: 48MP\n- Siêu rộng: 12MP\n- Tele: 12MP (Zoom
quang học 5x)\n- Cảm biến LiDAR\n- Camera trước:
12MP\n- Pin: Khoảng 4.422 mAh\n- Kết nối: USB-C,
hỗ trợ sạc nhanh và sạc không dây\n- Hệ điều hành:
ios 17 (hoặc mới hơn)\n- Kháng nước, bụi: IP68 ",
6     "price": "28990000.00",
7     "stock_quantity": 34,
8     "image_url": "https://product.hstatic.net/200000409445
Response Chart
```

2. Tìm kiếm sản phẩm (GET /products)

Nhập từ khóa bàn phím



GET Send

Query Headers Auth Body Tests Pre Run

Query Parameters

<input checked="" type="checkbox"/>	search	bàn phím
<input type="checkbox"/>	parameter	value

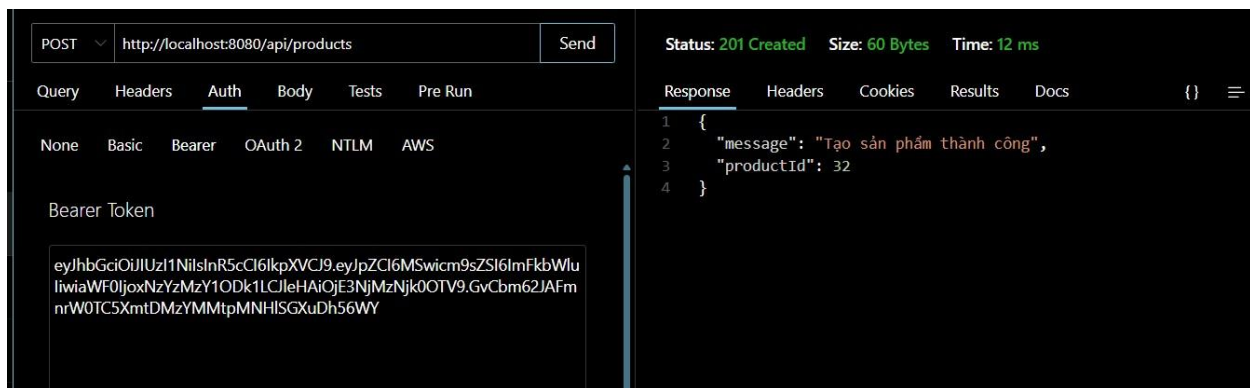
Status: 200 OK Size: 535 Bytes Time: 6 ms

Response Headers Cookies Results Docs {} ≡

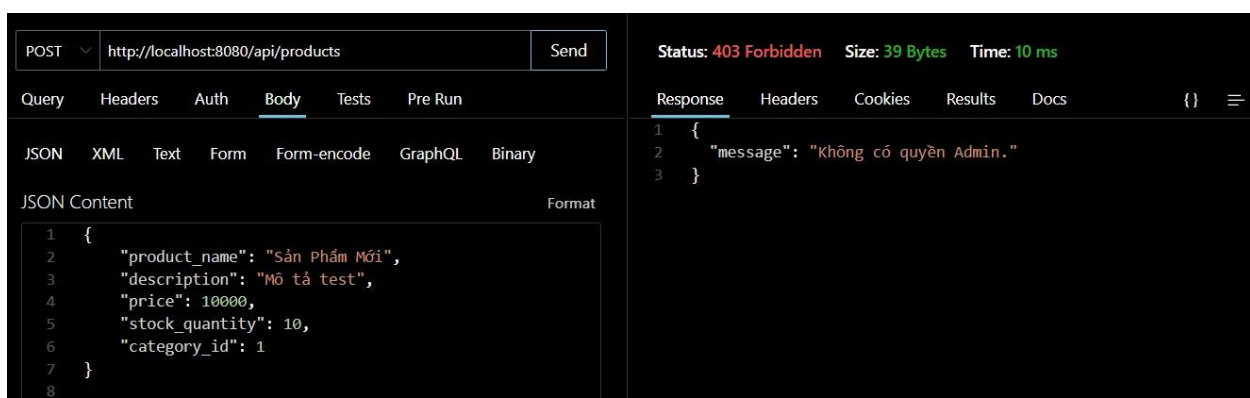
```
1 [
2   {
3     "product_id": 7,
4     "product_name": "Bàn phím cơ Aula F75",
5     "description": "THÔNG TIN SẢN PHẨM Aula F75\n- LAYOUT
75%\n- MẠCH XUÔI \n- LED RGB \n- KEYCAP PBT\n-
GASKET MOUNT\n- PIN 4000 MAH\n- 3 MODE KẾT NỐI (TYPE
C , BT 5.0, 2.4GHZ) \n- CỐ APP CHÍNH LED ĐI KÈM\n-
FULL FOAM\n- Switch : 2 loại Star Vector( Linear ) ,
swtich silent Outemu V2",
6     "price": "999000.00",
7     "stock_quantity": 14,
8     "image_url": "https://bizweb.dktcdn.net/thumb/1024x1024
/100/438/322/products/main-07.jpg?v=1761806374973",
9     "category_name": "Laptop & PC"
10  }
11 ]
Response Chart
```

3. Thêm sản phẩm mới vào danh sách sản phẩm (POST /products)

Chỉ có thể thêm sản phẩm mới khi là admin



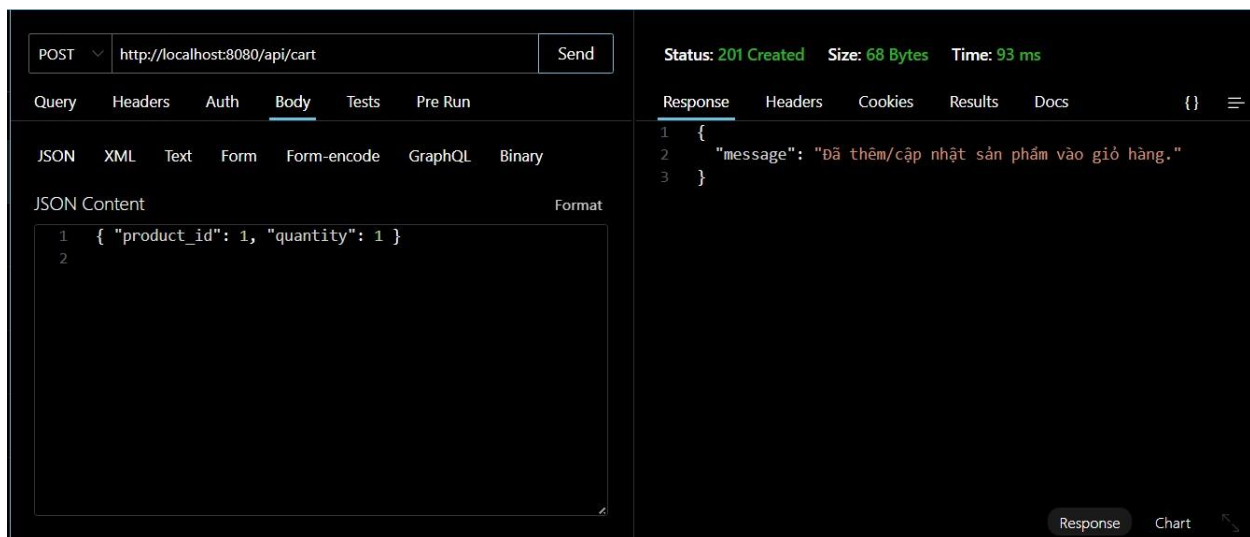
Trường hợp không phải là admin



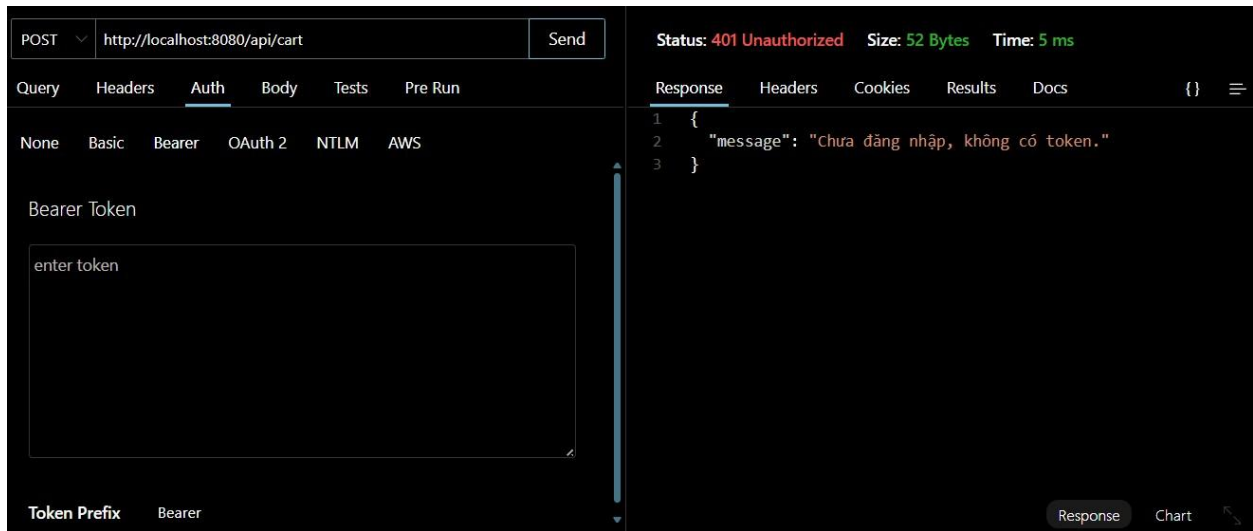
❖ Cart (Quản lý giỏ hàng)

1. Thêm sản phẩm vào giỏ (POST /cart)

Chỉ có thể thêm sản phẩm vào giỏ sau khi người dùng đã đăng nhập



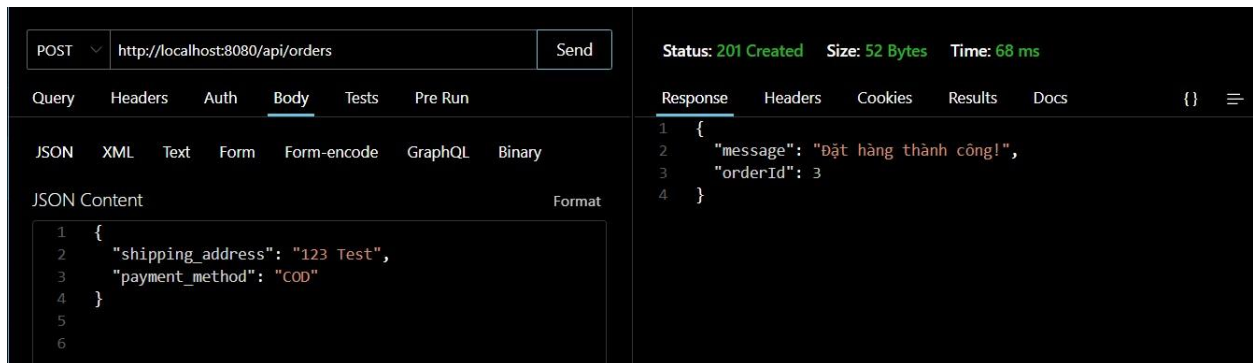
Trường hợp chưa đăng nhập



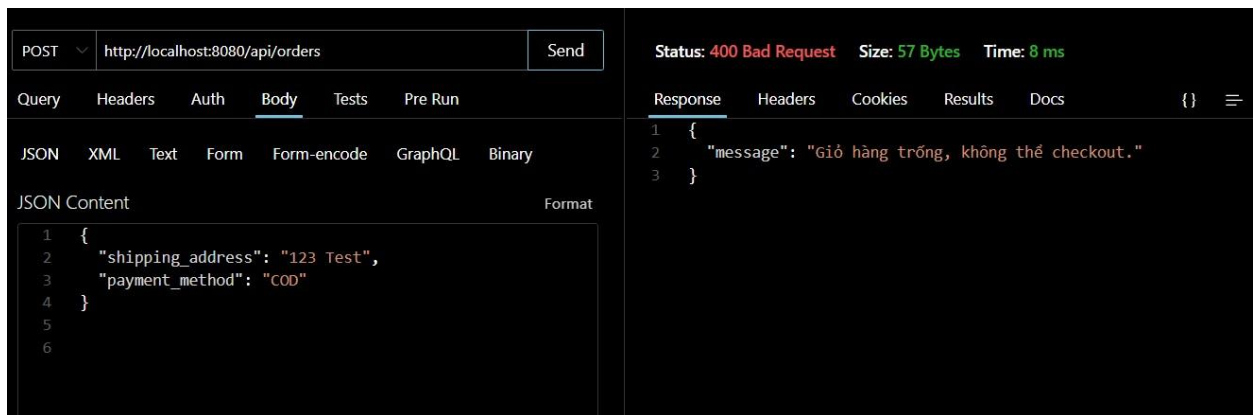
❖ Orders (Quản lý đơn hàng)

1. Checkout (POST /orders)

Chỉ có thể checkout khi giỏ hàng có ít nhất 1 vật phẩm trở lên

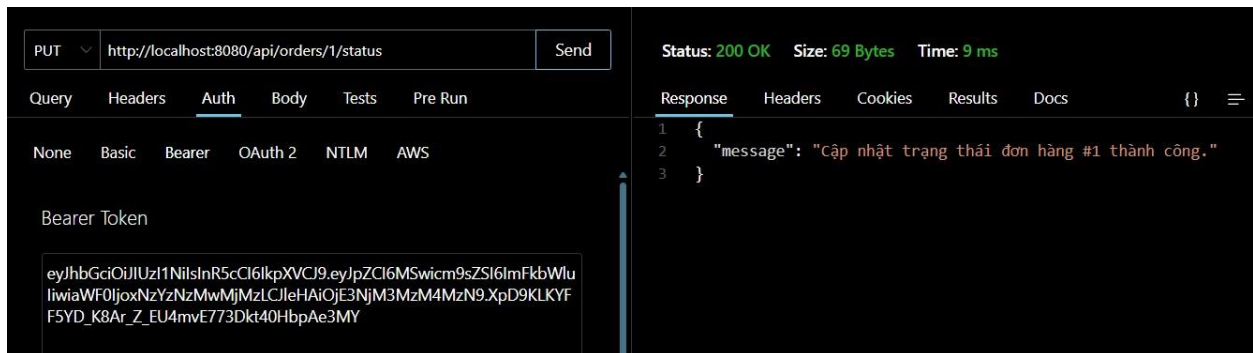


Trường hợp giỏ hàng rỗng

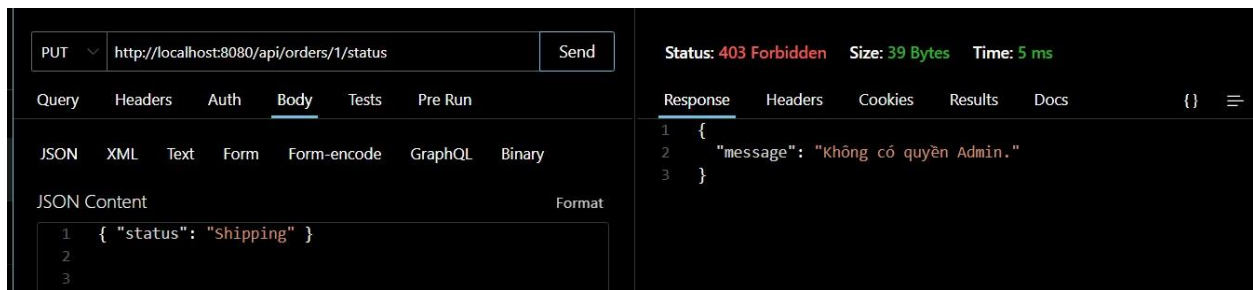


2. Đổi status đơn hàng (PUT /status)

Chỉ có thể cập nhật status khi là admin



Trường hợp không phải là admin



2. Báo cáo tỷ lệ bao phủ kiểm thử

- **Công cụ:** Jest Coverage
- **Hình ảnh:**

```
PS C:\Users\huynh> cd "C:\Users\huynh\OneDrive\Desktop\Programming\WEBSITE CNPM\Backend"
PS C:\Users\huynh\OneDrive\Desktop\Programming\WEBSITE CNPM\Backend> npm test -- --coverage

> backend@1.0.0 test
> jest --coverage

PASS tests/orderUtils.test.js
  Unit Test: Hàm calculateTotal (Tính tổng tiền)
    ✓ TC-01: Tính đúng tổng tiền cho giỏ hàng có 2 món (3 ms)
    ✓ TC-02: Trả về 0 nếu giỏ hàng rỗng
    ✓ TC-03: Báo lỗi nếu dữ liệu sai (Giá âm) (10 ms)
  Unit Test: Hàm validateOrderInput (Kiểm tra đầu vào)
    ✓ TC-04: Báo lỗi nếu thiếu địa chỉ giao hàng (1 ms)
    ✓ TC-05: Hợp lệ nếu đủ thông tin và đúng phương thức

File                | % Stmts | % Branch | % Funcs | % Lines | Uncovered Line #s
-----|-----|-----|-----|-----|-----
All files            | 93.33   | 93.75    | 100     | 93.33   |
  orderUtils.js      | 93.33   | 93.75    | 100     | 93.33   | 34

Test Suites: 1 passed, 1 total
Tests:       5 passed, 5 total
Snapshots:   0 total
Time:        0.881 s, estimated 1 s
Ran all test suites.
```

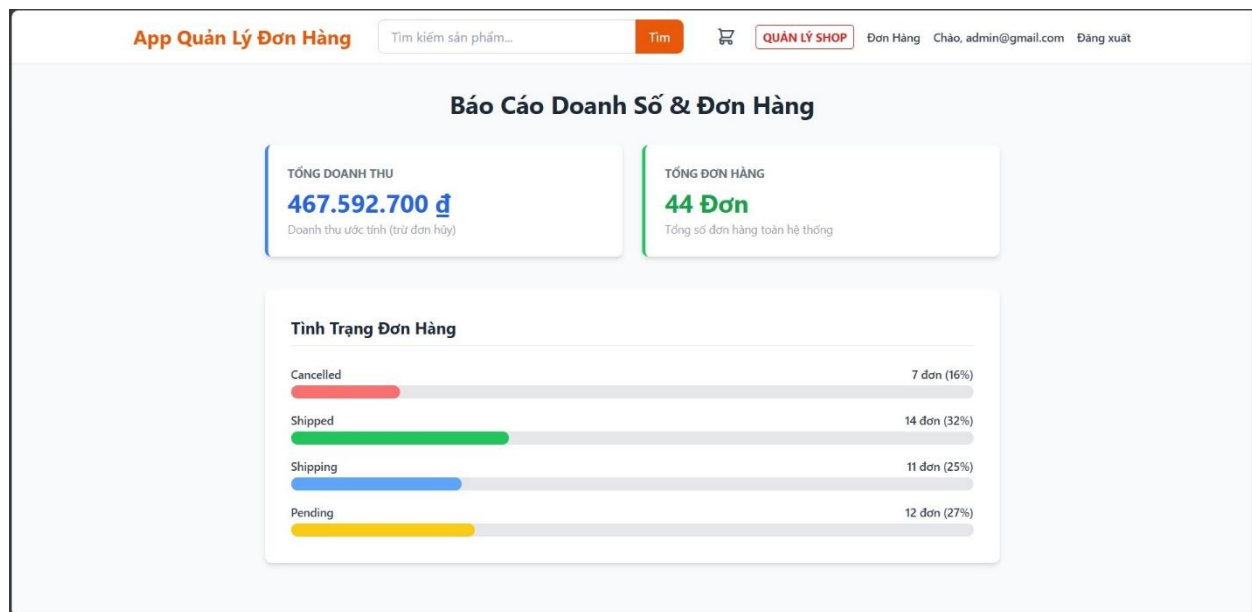


Kết quả:

- **Độ bao phủ hàm (Funcs): Đạt 100% (2/2 hàm).** Tất cả các hàm xử lý logic nghiệp vụ cốt lõi đều đã được đưa vào kịch bản kiểm thử.
- **Độ bao phủ câu lệnh (Stmts) và Dòng (Lines): Đạt 93.33%.** Hầu hết các dòng mã thực thi đều đã được kiểm tra tính đúng đắn.
- **Độ bao phủ nhánh (Branch): Đạt 93.75%.** Các trường hợp điều kiện biên, các luồng if-else xử lý lỗi dữ liệu đầu vào đều đã được bao quát.

Kết luận: Mã nguồn đạt chất lượng cao, đảm bảo tính ổn định và giảm thiểu tối đa các lỗi logic trong quá trình vận hành thực tế (như tính sai tiền, lỗi dữ liệu đầu vào).

3. Báo cáo doanh số, trạng thái bán hàng



4. Đề xuất cải tiến

4.1. Tích hợp thanh toán thực tế

- Tích hợp API của VNPAY hoặc Momo.

- Xây dựng hệ thống IPN (Instant Payment Notification) để nhận phản hồi tự động từ ngân hàng khi khách chuyển khoản thành công, thay vì xác nhận thủ công.

4.2. Tích hợp vận chuyển

- Hiện tại: trạng thái đơn hàng được cập nhật thủ công bởi Admin.
- Cải tiến: Kết nối API với Giao Hàng Nhanh (GHN) hoặc Giao Hàng Tiết Kiệm (GHTK) để đồng bộ trạng thái đơn hàng theo thời gian thực