TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO GIỮA KÌ**

**CHUYÊN ĐỀ CÔNG NGHỆ PHẦN MỀM**

**TÌM HIỂU WEB API VÀ DEMO**

*Người hướng dẫn*: **GV LỤC MINH TUẤN**

*Người thực hiện*: **NGÔ MINH HIẾU – 51702017**

Lớp **: 17050201**

Khoá  **: 21**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2021**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO GIỮA KÌ**

**CHUYÊN ĐỀ CÔNG NGHỆ PHẦN MỀM**

**TÌM HIỂU WEB API VÀ DEMO**

Người hướng dẫn: **GV LỤC MINH TUẤN**

Người thực hiện: **NGÔ MINH HIẾU**

Lớp **: 17050201**

Khoá  **: 21**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2021**

**ĐỒ ÁN ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là sản phẩm đồ án của riêng tôi và được sự hướng dẫn của GV Lục Minh Tuấn ;. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày 16 tháng 11 năm 2021*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Hieu*

*Ngô Minh Hiếu*

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

**Phần xác nhận của GV hướng dẫn**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

**Phần đánh giá của GV chấm bài**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

MỤC LỤC

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN 2](#_Toc89476413)

[MỤC LỤC 3](#_Toc89476414)

[CHƯƠNG 1 – GIỚI THIỆU VỀ WEB API 4](#_Toc89476415)

[CHƯƠNG 2 – THỰC HIỆN XÂY DỰNG WEB API VỚI C# 7](#_Toc89476416)

[Hình 1: API Lấy tất cả sản phẩm 8](#_Toc89476868)

[Hình 2: API Lấy chi tiết sản phẩm 9](#_Toc89476869)

[Hình 3 : API Lấy chi tiết sản phẩm không tồn tại 9](#_Toc89476870)

[Hình 4 : API dùng để đăng nhập 10](#_Toc89476871)

[Hình 5: API dùng để đăng kí 11](#_Toc89476872)

[Hình 6 : API dùng để thêm sản phẩm 12](#_Toc89476873)

[Hình 7: API trả kết quả thêm sản phẩm thất bại 13](#_Toc89476874)

[Hình 8: API dùng để xóa sản phẩm 13](#_Toc89476875)

[Hình 9: API trả kết quả việc xóa sản phẩm 14](#_Toc89476876)

[Hình 10: API để sửa đổi thôn tin sản phẩm 15](#_Toc89476877)

[Hình 11: API dùng thay đổi thông tin tài khoản 16](#_Toc89476878)

[Hình 12: API thay đổi mật khẩu của tài khoản 17](#_Toc89476879)

[Hình 13: API lấy sản phẩm theo ID đơn hàng 18](#_Toc89476880)

[Hình 14: API lấy tất cả đơn hàng theo ID của user 19](#_Toc89476881)

[Hình 15: API dùng để thêm sản phẩm vào đơn hàng 20](#_Toc89476882)

[Hình 16: API dùng để xóa đơn hàng 21](#_Toc89476883)

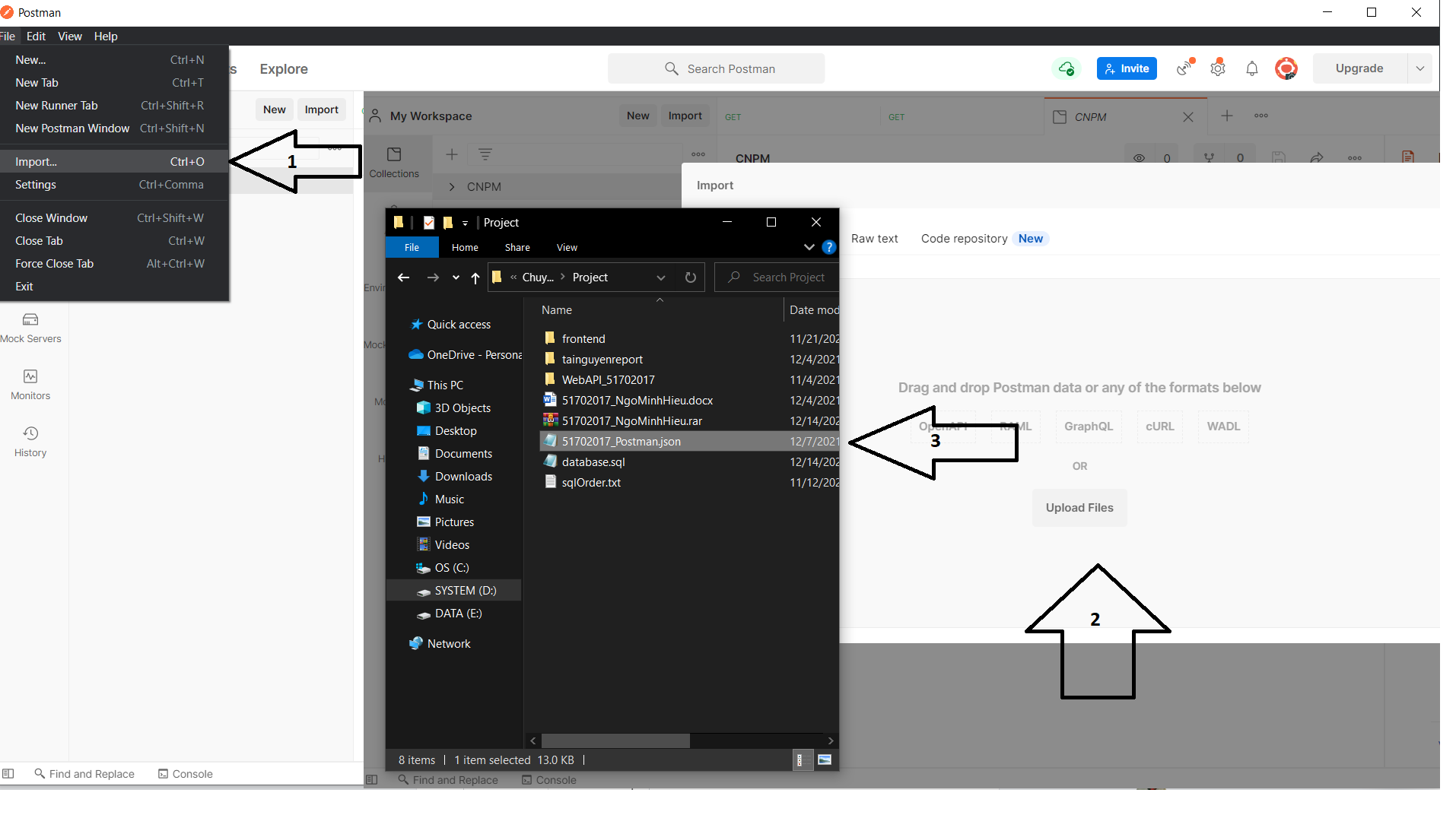
[Hình 17: API dùng để xóa sản phẩm trong đơn hàng 22](#_Toc89476884)

CHƯƠNG 1 – GIỚI THIỆU VỀ WEB API

* API (Application Programming Interface) là phương thức, giao thức kết nối với các thư viện hoặc một ứng dụng nào khác. API cung cấp khả năng truy xuất đến một tập dữ liệu thông qua một hoặc mốt số hàm được thiết kế ra và có thể trao đổi dữ liệu giữ các ứng dụng.
* Web API được sử dụng trong trong các hệ thống website. Có một số website đều được thiết kế theo dạng web API cho phép người lập trình viên hiểu và có thể sử dụng để gửi yêu cầu đến web API đó để có thể lấy dữ liệu phù hợp với yêu cầu request mà web API đó đặt ra. Ví dụ như ứng dụng của bạn được viết ra với các yêu cầu từ khách hàng là có thể sử dụng tài khoản có sẵn thuộc một số trang mạng xã hội, trang công nghệ lớn như facebook , google , gmail, … yêu cầu người lập trình viên phải đọc cách giao tiếp với các API trên để cấp quyền cũng như đảm bảo API của bạn gọi đến API của họ một cách hợp lệ.
* Hệ thống API được thiết kế tuân theo tiêu chuẩn REST và HTTP như các phương thức sau
  + GET : được dùng để lấy dữ liệu từ API
  + POST : được dùng để đưa dữ liệu đến API
  + PUT : được dùng để đưa dữ liệu đến API và thường được dùng cập nhật lại dữ liệu đã (quy ước chung theo tiêu chuẩn REST)
  + DELETE: như tên của phương thức thì phương thức này sử dụng để gửi yêu cầu đến API xóa một thông tin nào đó có trong API khi bạn có quyền được thực hiện việc xóa dữ liệu trên API đó
* Web API hoạt động bằng cách chọn một ngôn ngữ như c# , js , java…. Để tiến hành xây dựng Web API thì tùy theo ngôn ngữ và công nghệ chọn sử dụng để viết ở phía web API mà có các đường dẫn ( URL ) API được sử dụng để người phát triển dùng để gọi đến Web qua các công cụ được thiết kế để test API như Postman , insomnia,…. Hoặc nếu bạn kết hợp giữa web API và một framework của js về front-end như Vuejs, Reactjs, …. Để gọi dữ liệu và hiện thị dữ liệu được lấy từ web API.
* Server sẽ trả về dựa theo sự thiết kế của người phát triển và thường quy ước chung trong việc viết API là trả về dưới dạng json hoặc XML thông qua giao thức HTTP hoặc HTTPS
* Ở phía được gọi là Client như ứng dụng gửi request hay là phía giao diện được viết bằng framework thì sẽ sử dụng kĩ thuật parse dữ liệu json sang dữ liệu và bóc tách dữ liệu đó để hiện thị ở những nơi mà người phát triển muốn hiện trên trình duyệt web hoặc là ứng dụng di động, còn đối với các phần mềm gửi request thì ta đọc dữ liệu dưới dạng json để biết việc gửi request và việc trả kết quả của server được viết có đạt yêu cầu hay không.
* Ưu điểm của web API:
  + Web API được sử dụng hầu hết trên các ứng dụng desktop, ứng dụng mobile và ứng dụng website.
  + Linh hoạt với các định dạng dữ liệu khi trả về client: Json, XML hay định dạng khác.
  + Nhanh chóng xây dựng HTTP service: URI, request/response headers, caching, versioning, content formats và có thể host trong ứng dụng hoặc trên IIS.
  + Mã nguồn mở, hỗ trợ chức năng RESTful đầy đủ, sử dụng bởi bất kì client nào hỗ trợ XML, Json.
  + Hỗ trợ đầy đủ các thành phần MVC như: routing, controller, action result, filter, model binder, IoC container, dependency injection, unit test.
  + Giao tiếp hai chiều được xác nhận trong các giao dịch, đảm bảo độ tin cậy cao.
* Nhước điểm :
  + Do web API còn khá mới nên chưa thể đánh giá nhiều về nhược điểm của mô hình nay. Tuy nhiên, có hai nhược điểm dễ dàng nhận thấy:
  + Web API chưa hoàn toàn phải là RESTful service, mới chỉ hỗ trợ mặc định GET, POST
  + Để sử dụng hiệu quả cần có kiến thức chuyên sâu, có kinh nghiệm backend tốt
  + Tốn thời gian và chi phí cho việc phát triển, nâng cấp và vận hành
  + Có thể gặp vấn đề về bảo mật khi hệ thống bị tấn công nếu không giới hạn điều kiện kỹ.

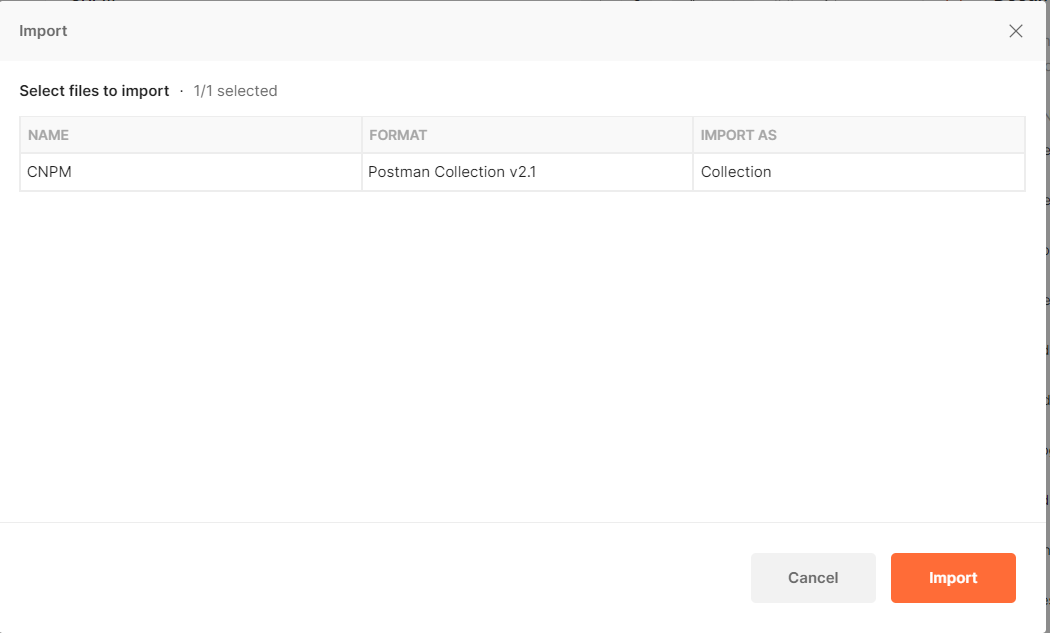
CHƯƠNG 2 – THỰC HIỆN XÂY DỰNG WEB API VỚI C#

1. Cách sử dụng SQL và POSTMAN



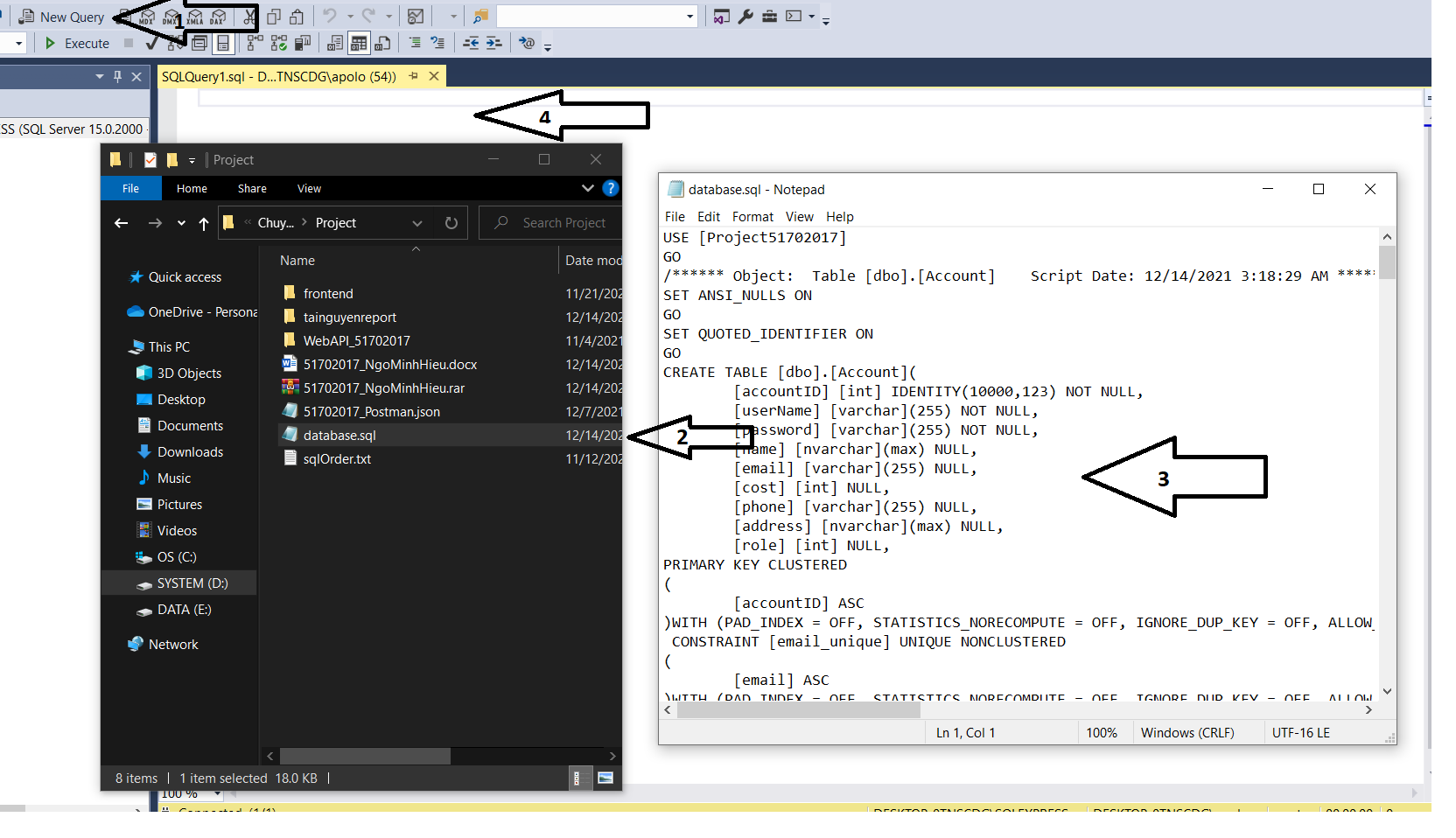
Hình : Cách import method vào postman

* Mở phần mềm POSTMAN
* 1 : Chọn file -> import
* 2 : Sẽ xuất hiện cửa sổ để import file vào
* 3 : Chọn file 51702017\_Postman.json để import vào



Hình : Cách import method vào postman

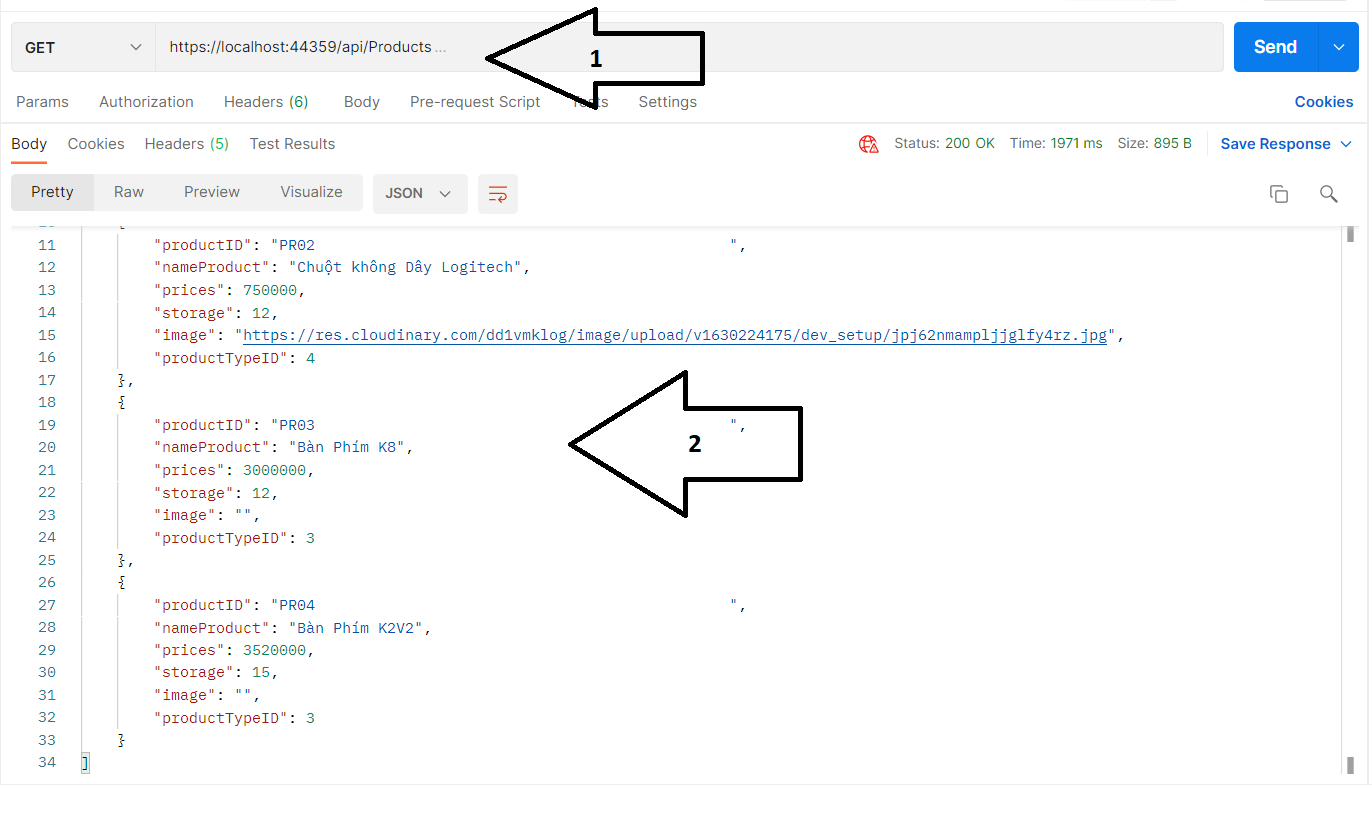
* Sau đó sẽ xuất hiện cửa sổ chọn import là sẽ xuất hiện các method để sử dụng với API



Hình : Cách sử dụng database để đưa dữ liệu vào SQL Server

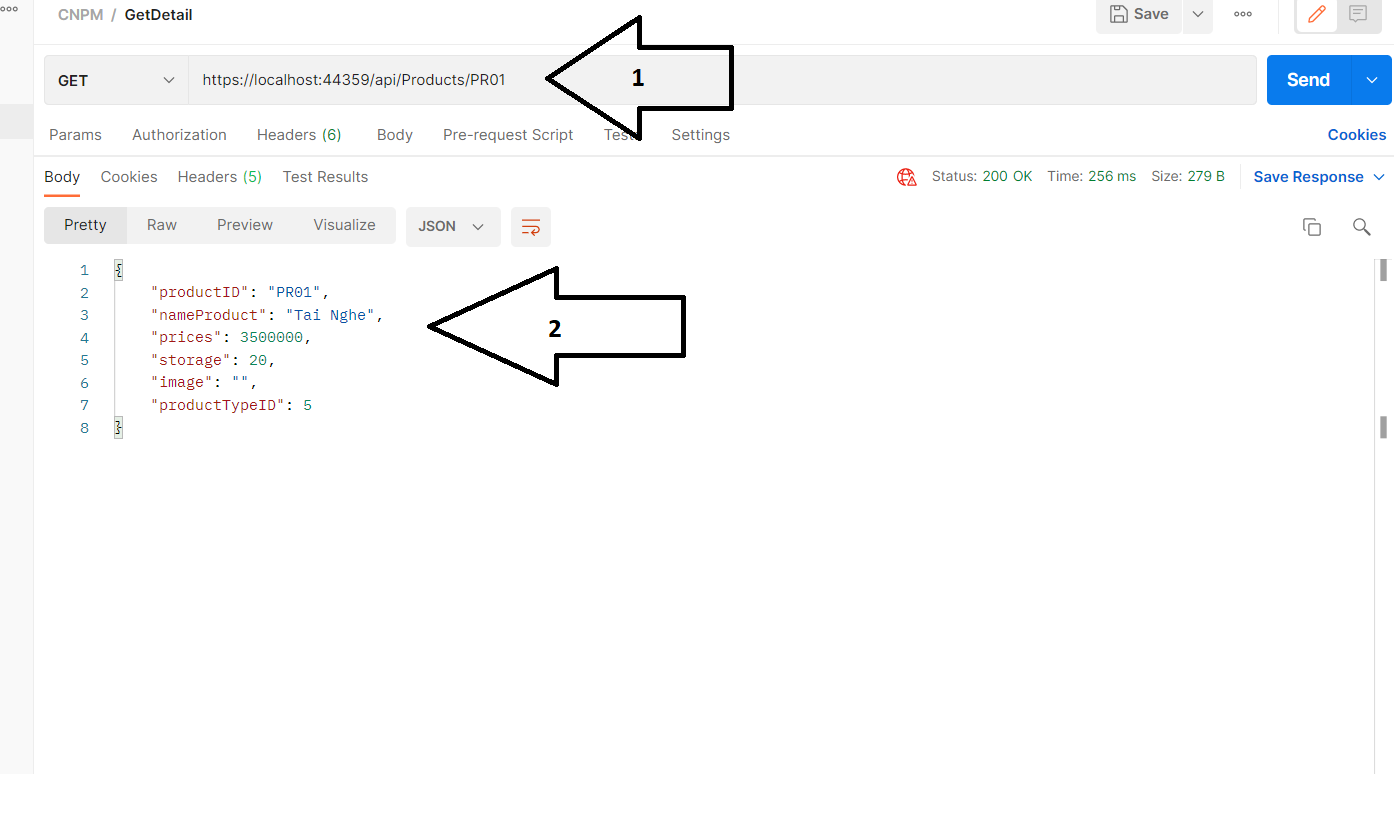
* 1 : Khởi động SQL Server Management Studio và chọn new query
* 2 : Mở file database.sql
* 3 : Copy tất cả các dòng lệnh trong file
* 4 : Past vào SQLQuery và chọn Excute tất cả dòng lệnh để khởi tạo database

1. Chi tiết cách sử dụng API



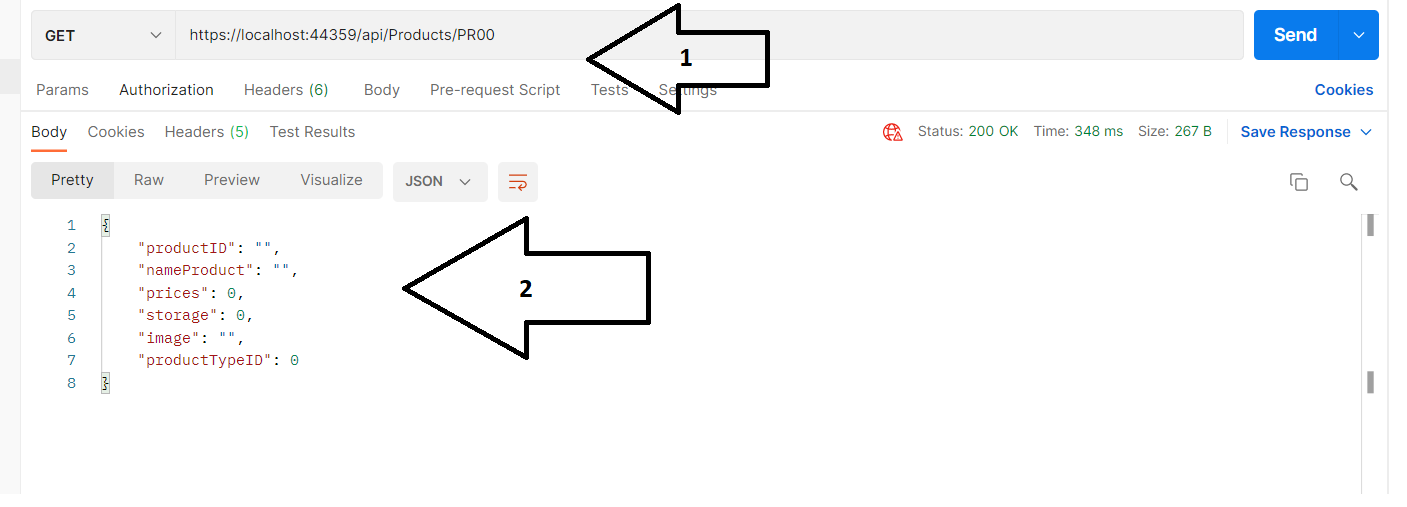
Hình 4: API Lấy tất cả sản phẩm

* 1 : Đường dẫn tới api : <https://localhost:44359/api/Products>
* 1 : Phương thức được sử dụng là GET
* 2 : Chức năng được dùng để lấy tất cả sản phẩm có trong cơ sở dữ liệu



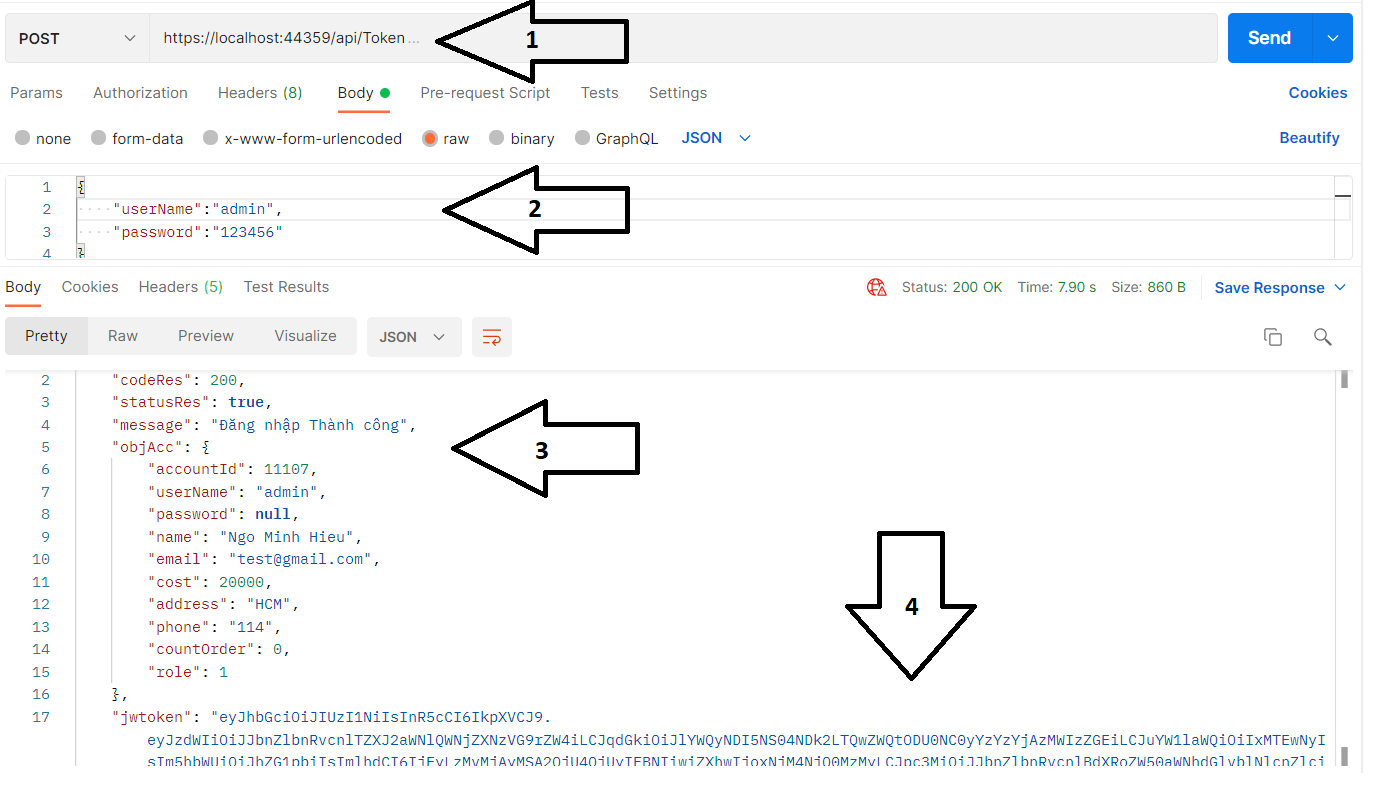
Hình 5: API Lấy chi tiết sản phẩm

* 1 : Đường dẫn tới api : [https://localhost:44359/api/Products/{idproduct}](https://localhost:44359/api/Products/%7bidproduct%7d)
* 1 : Phương thức được sử dụng là phương thức GET, dùng để lấy chi tiết sản phẩm
* 2 : Là dữ liệu được API trả về.



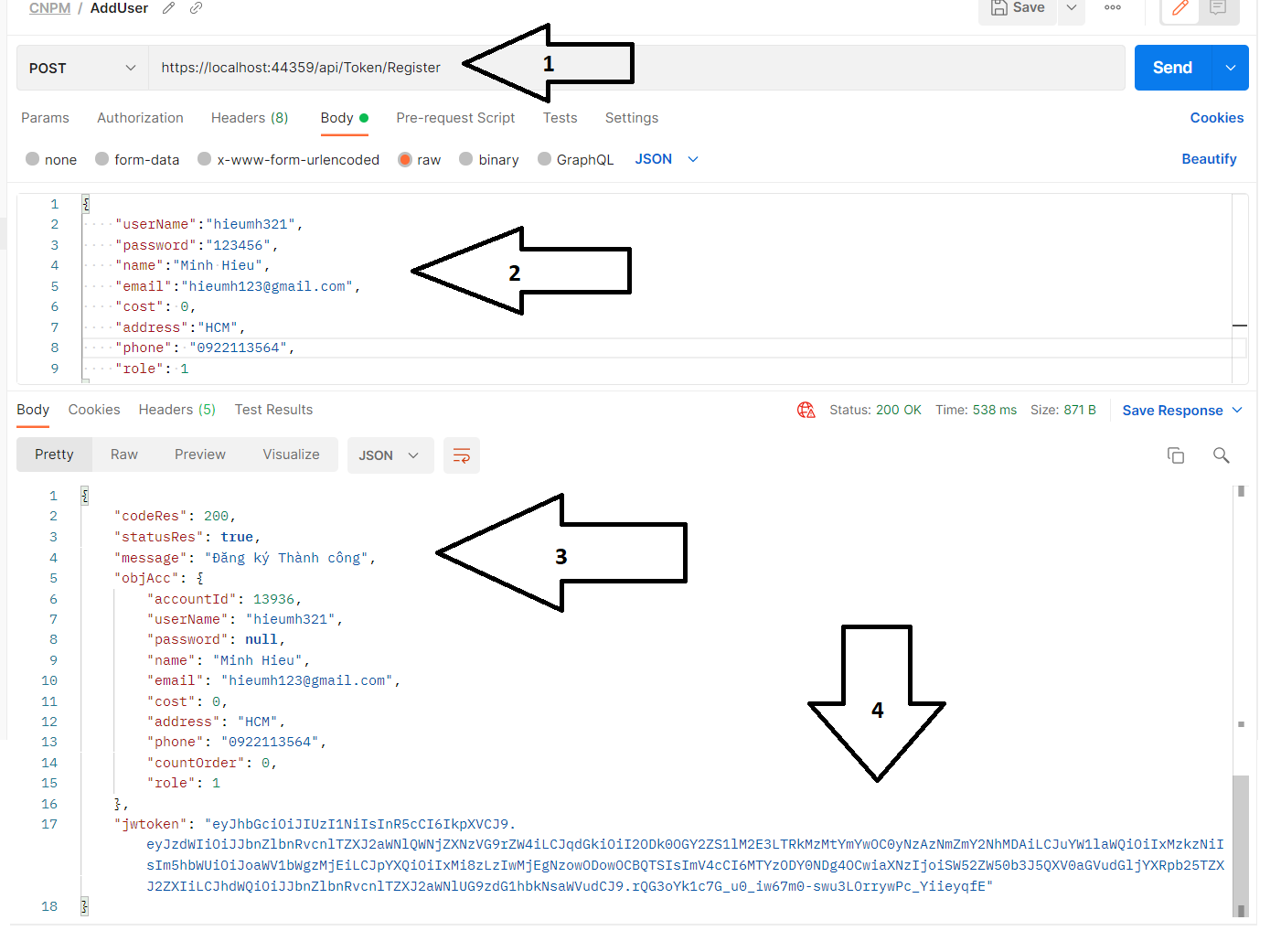
Hình 6 : API Lấy chi tiết sản phẩm không tồn tại

* Nếu id product (1) không nằm trong cơ sở dữ liệu thì phía web API sẽ trả về dữ liệu có dạng ProductId là dạng rỗng (2).



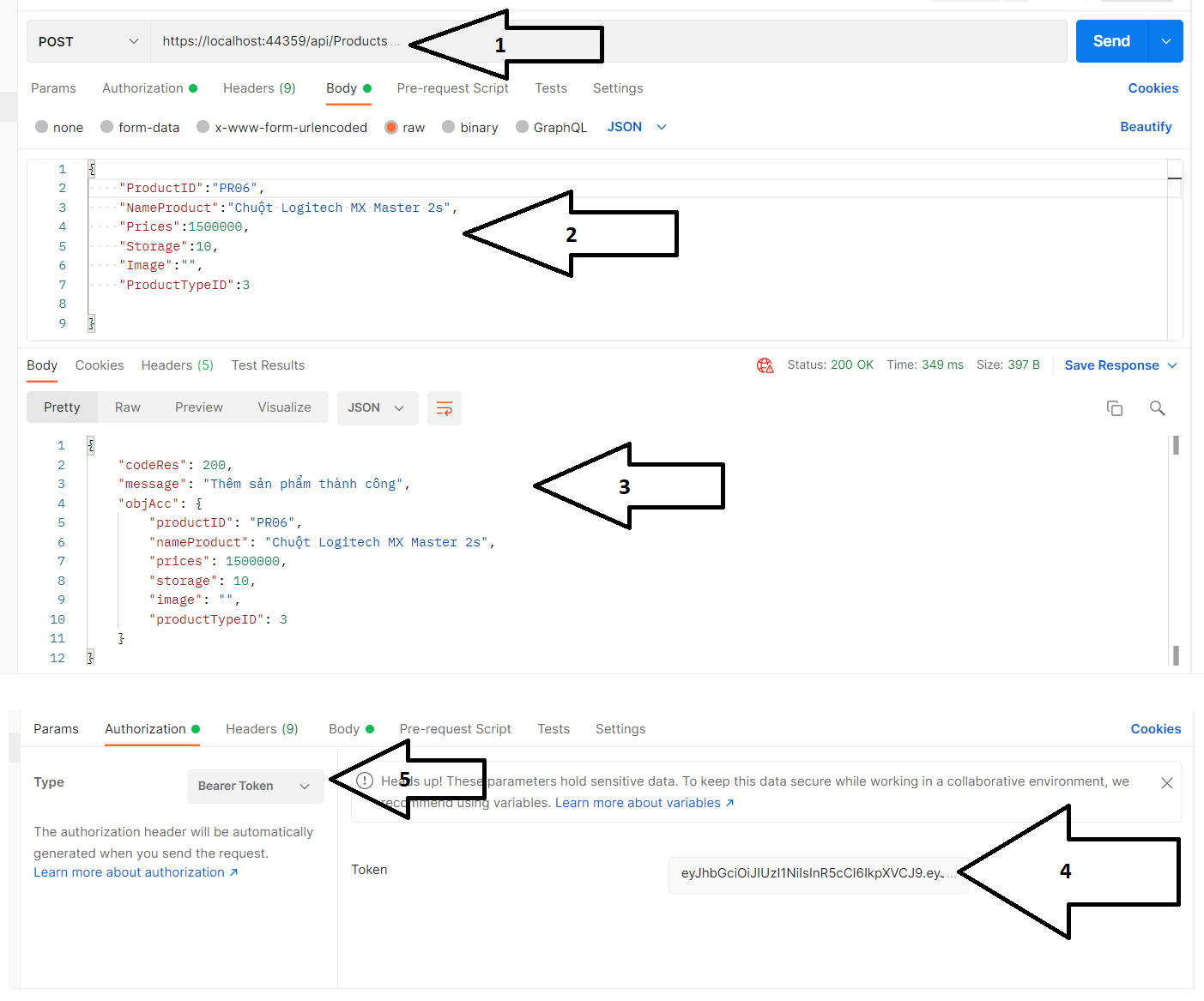
Hình 7 : API dùng để đăng nhập

* 1 : Đường dẫn tới api : <https://localhost:44359/api/Token>
* 1 : Phương thức được sử dụng gọi đến đường dẫn này là POST
* 2 : Thông tin truyền theo gồm userName , password
* 3 : Kết quả API trả về là thông tin tài khoản
* 4: Là JWT được đính kèm dùng để thao tác với các tác vụ khác
* Được dùng để đăng nhập.



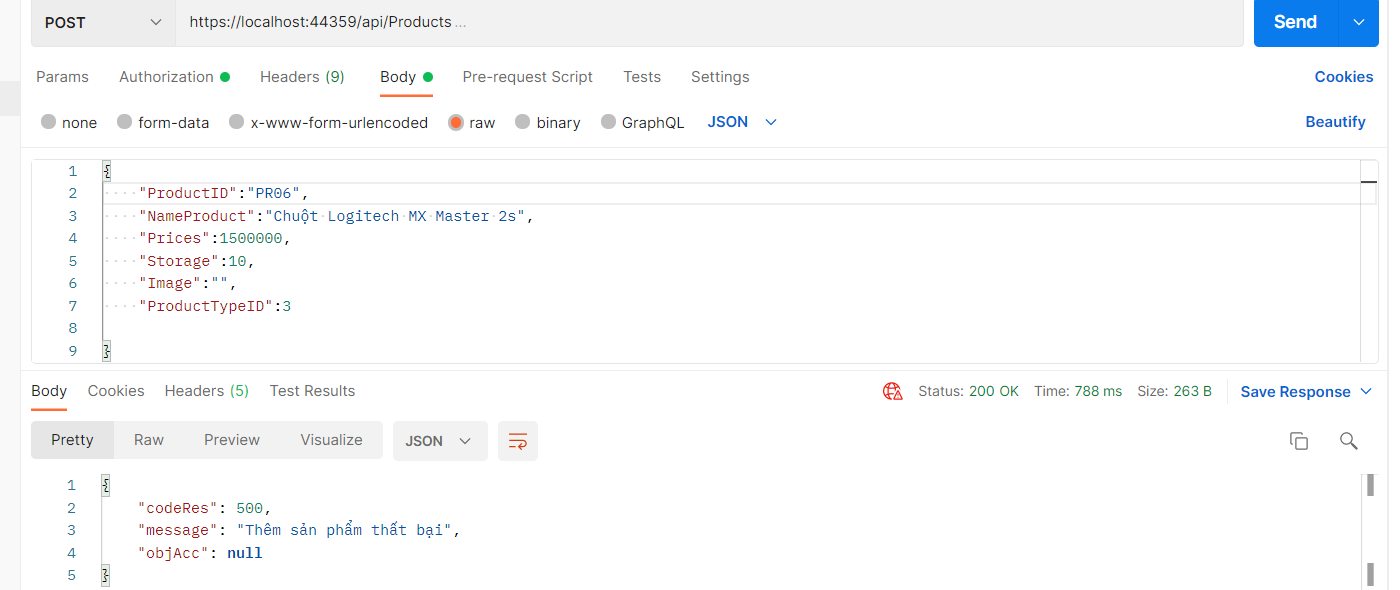
Hình 8: API dùng để đăng kí

* 1 : Đường dẫn tới api : <https://localhost:44359/api/Token>/Register
* 1 : Phương thức được sử dụng gọi đến đường dẫn này là POST
* 2 : Thông tin truyền theo gồm userName, password,name,email,address,phone và role sẽ được định nghĩa ở front-end gán cứng ở form để phân biệt quyền admin
* 3 : Kết quả API trả về là thông tin tài khoản đã đăng kí thành công
* 4: Là JWT được đính kèm dùng để thao tác với các tác vụ khác
* Được dùng để đăng nhập.



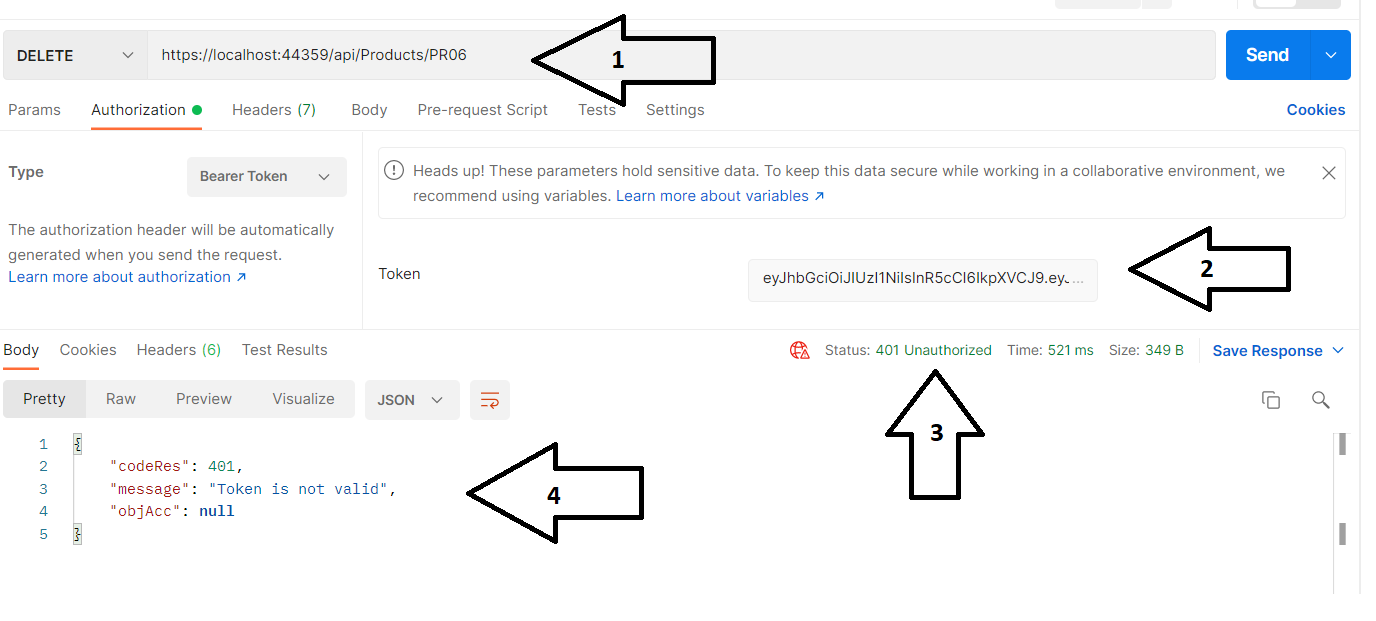
Hình 9 : API dùng để thêm sản phẩm

* 1 : Đường dẫn tới api : <https://localhost:44359/api/Products>
* 1 : Phương thức sử dụng ở đây là Post, được dùng để thêm sản phẩm.
* 2 : Thôngtin sản phẩm productid,nanmeProduct,prices,storage,imag,productid.
* 3 : Thông tin được trả về từ web API
* 4,5 : Sau khi đăng nhập ta có jwt từ đó ta gán cùng theo request với bearer token gửi đến server web api.



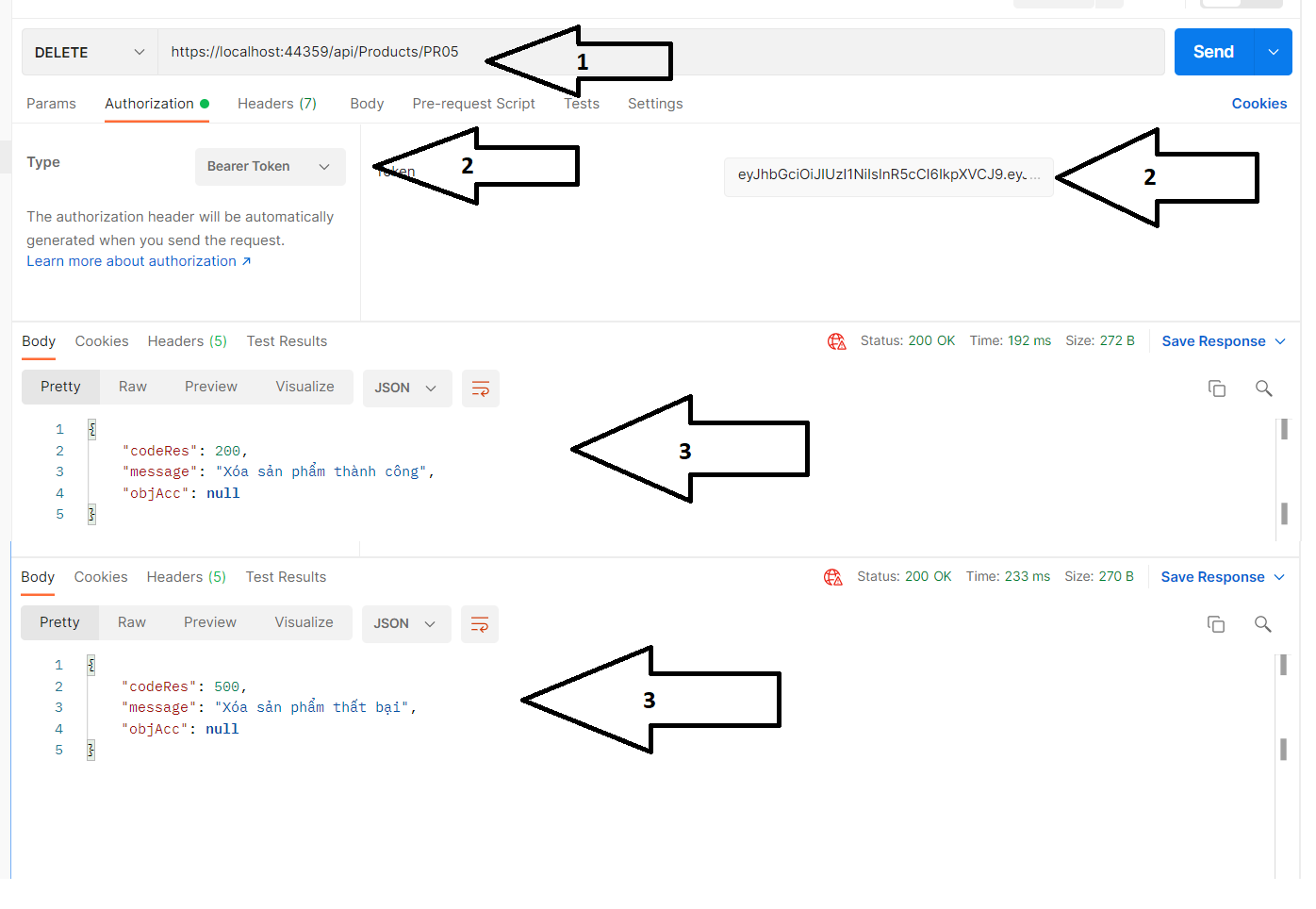
Hình 10: API trả kết quả thêm sản phẩm thất bại

* Nếu ta tiến hành thêm 1 sản phẩm y chan lần trước thì server trả về thêm sản phẩm thất bị vì ProductId là duy nhất và không được trùng.

****

Hình 11: API dùng để xóa sản phẩm

* 1 : Đường dẫn tới api : [https://localhost:44359/api/Products/{idproduct}](https://localhost:44359/api/Products/%7bidproduct%7d)
* 1 : Phương thức được sử dụng là phương thức Delete được dùng để xóa sản phẩm
* 2 : Đính kèm jwt sau khi đăng nhập vào bearer Token
* 3 : Code mặc định của web API trả về qua là 401 unauthorzied
* 4 : dữ liệu được web API trả về dưới dạng json để tiện xử lý ở phía client với các thông tin là codeRes, message , objecAcc(tùy biến theo một số phương thức)

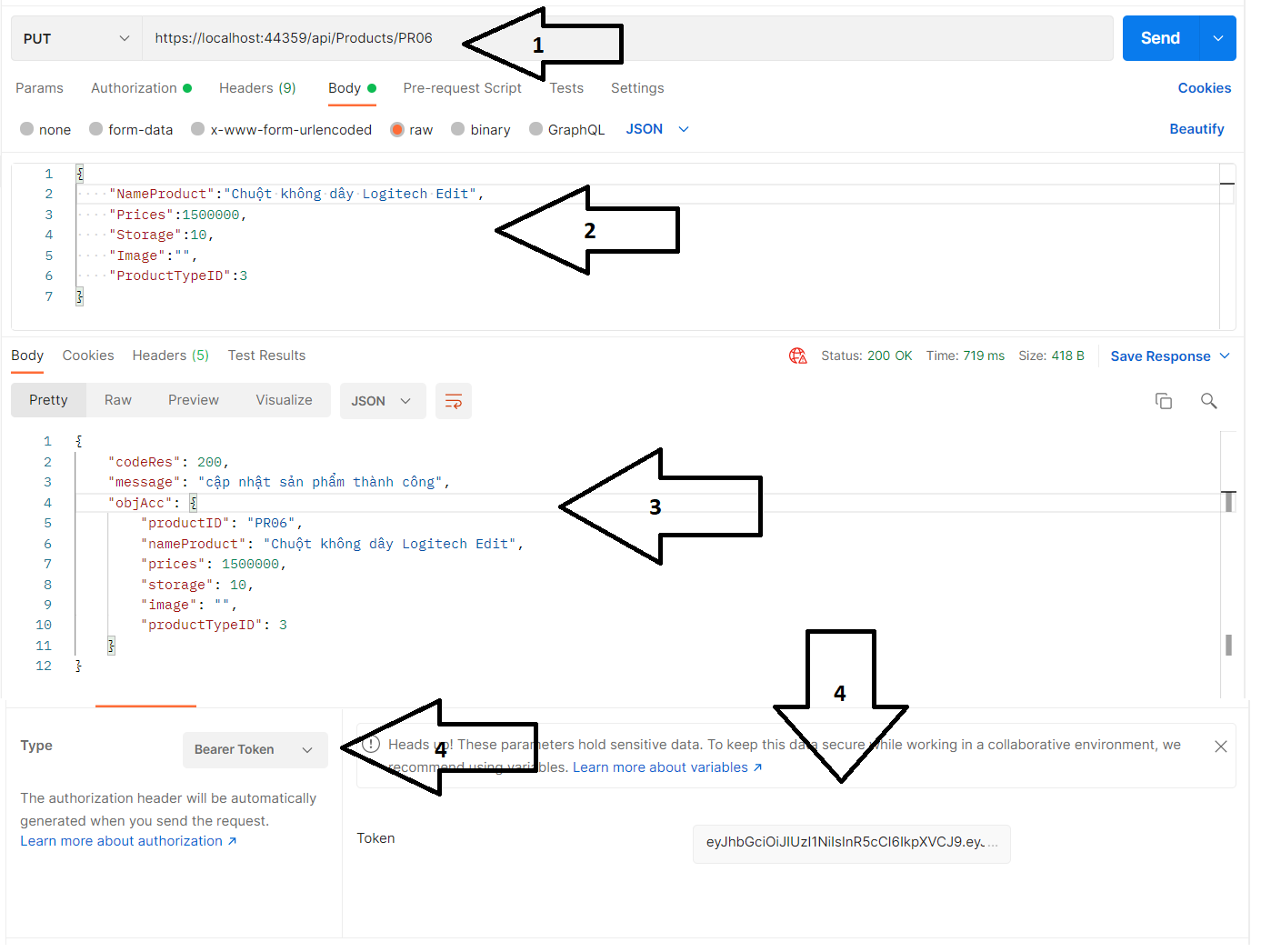
****

Hình 12: API trả kết quả việc xóa sản phẩm

* Nếu kèm theo token hợp lệ thì thực hiện yêu cầu xóa thành công API sẽ trả về

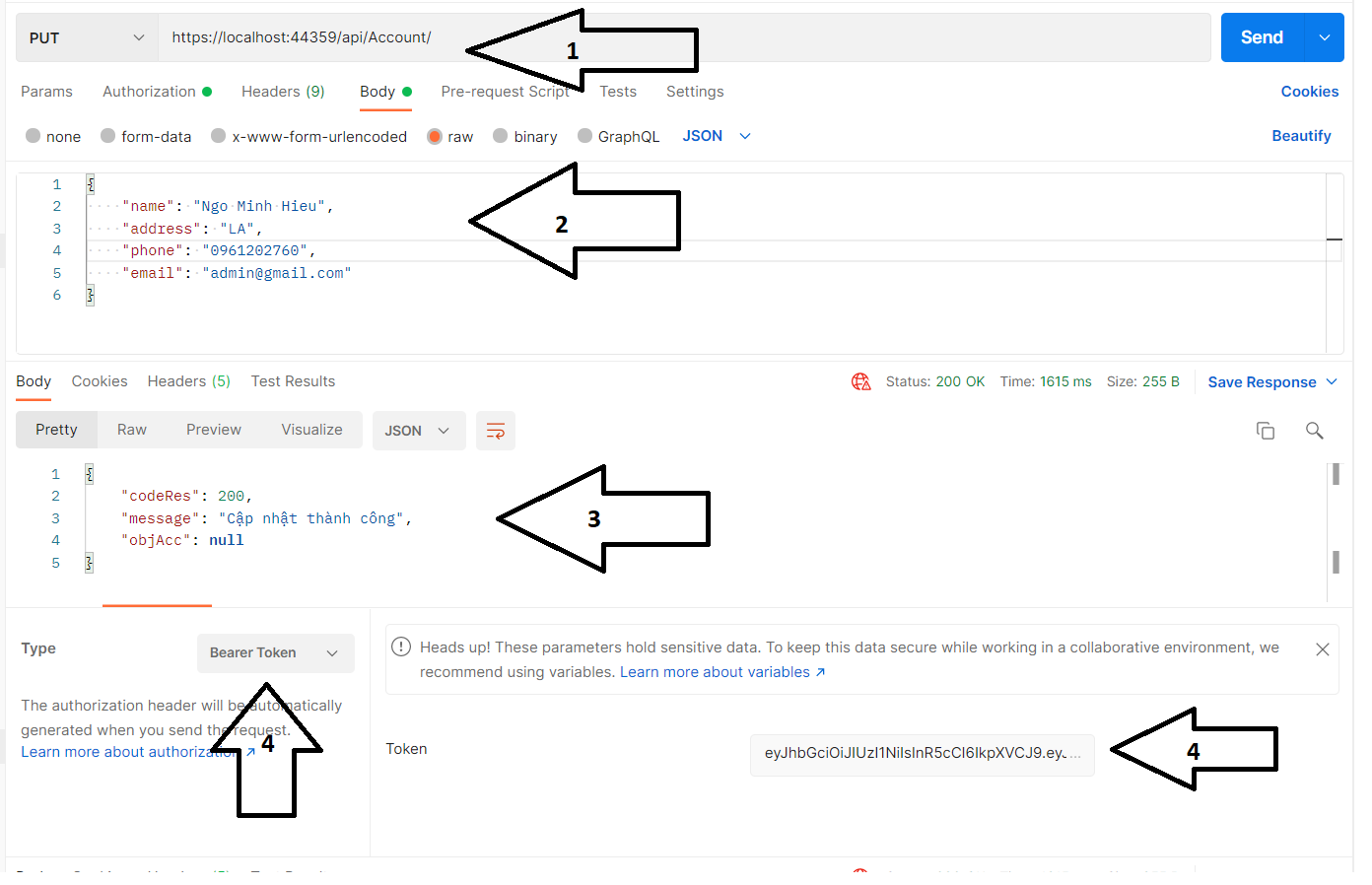
dữ liệu như hình và kết quả là thực hiện xóa sản phẩm thành công .

Nếu thực hiện việc xóa với cùng một mã sản phẩm thì sẽ báo lỗi là xóa thất bại vì đã xóa mất sản phẩm có mã như trước

****

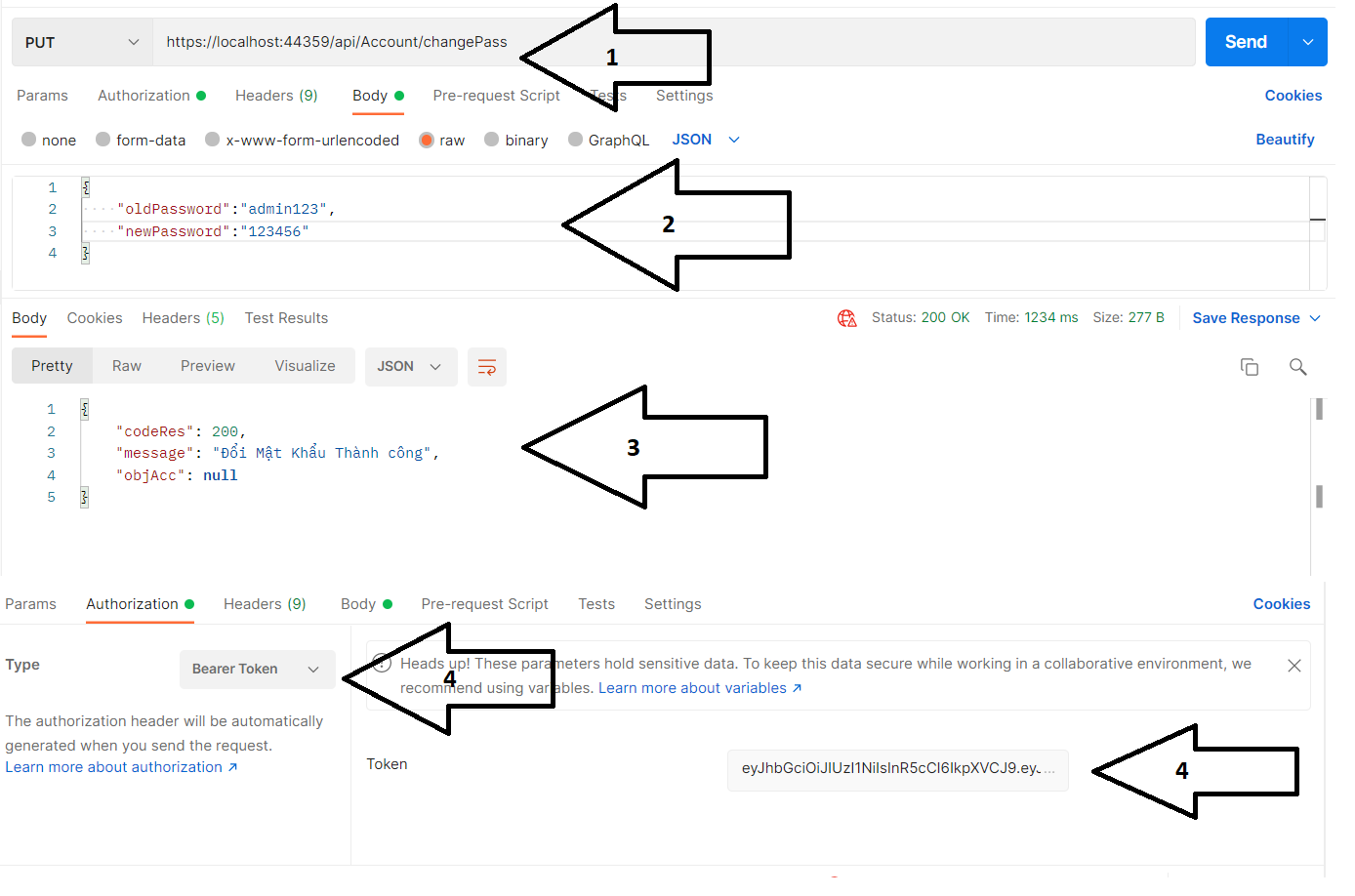
Hình 13: API để sửa đổi thôn tin sản phẩm

* 1 : Đường dẫn tới api : [https://localhost:44359/api/Products/{idproduct}](https://localhost:44359/api/Products/%7bidproduct%7d)
* 1 : Phương thức được sử dụng là phương thức PUT
* 2 : Thông tin được gửi đi kèm gồm nanmeProduct,prices,storage,imag,productid.
* 3 : Thông tin được trả về nếu cập nhật thành công
* 4 : Đính kèm jwt vào bearer token để request gửi lên web api có thể thực hiện được thao tác cập nhật

****

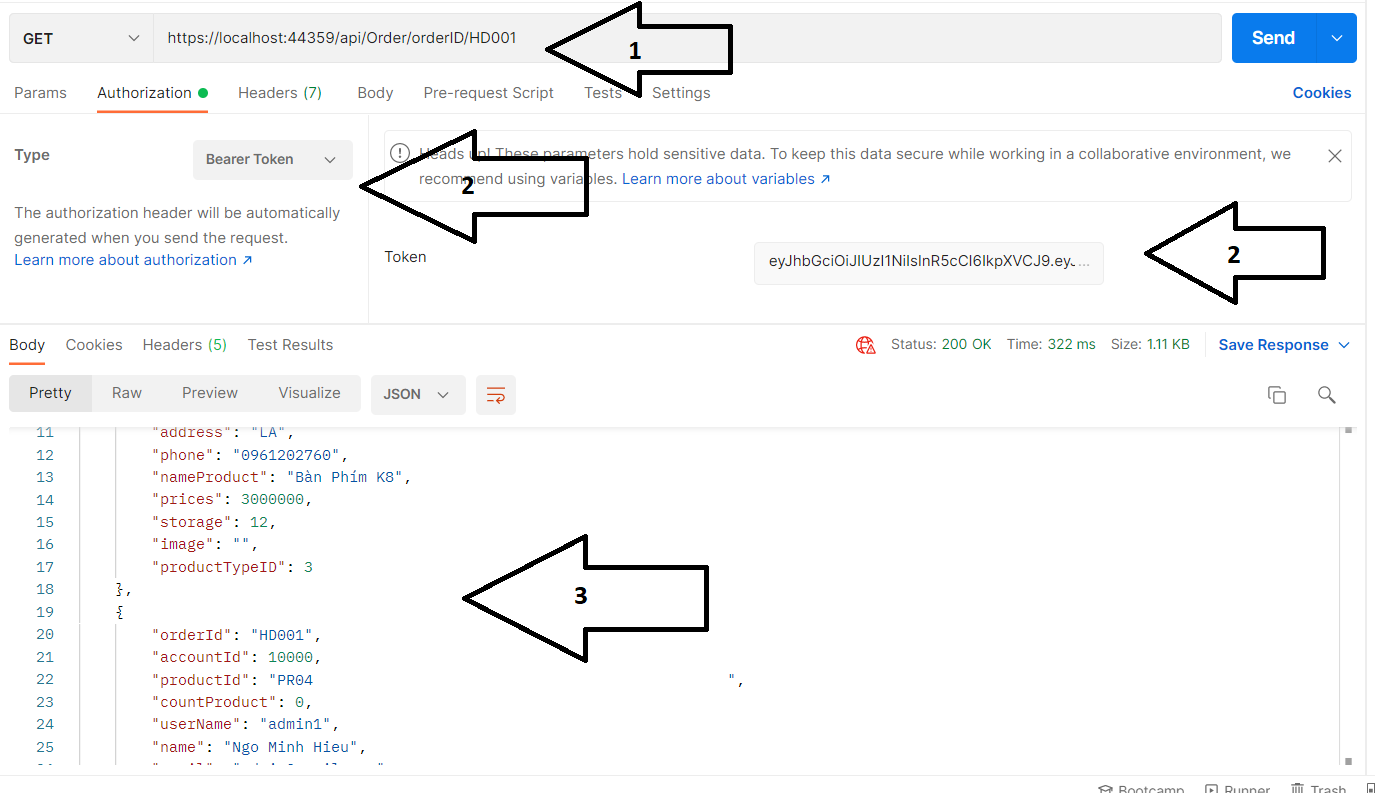
Hình 14: API dùng thay đổi thông tin tài khoản

* 1 : Đường dẫn tới api : <https://localhost:44359/api/>Account
* 1 : Phương thức được sử dụng là phương thức PUT
* 2 : Thông tin được gửi đi kèm gồm name, address , phone, email.
* 3 : Thông tin được trả về nếu cập nhật thành công
* 4 : đính jwt vào bearer token để lấy được userid trong jwt.



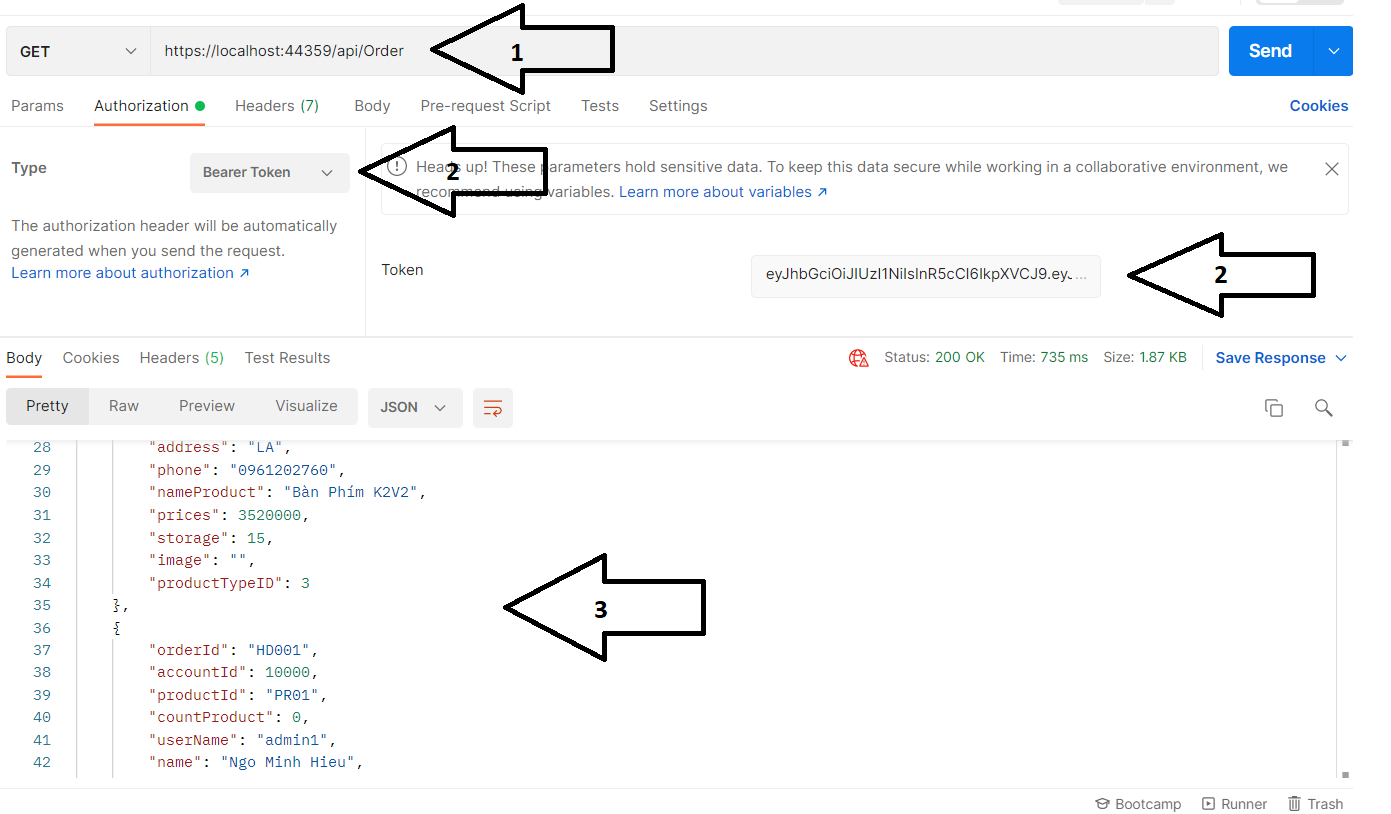
Hình 15: API thay đổi mật khẩu của tài khoản

* 1 : Đường dẫn tới api : <https://localhost:44359/api/Account/changePass>
* 1 : Phương thức được sử dụng là phương thức PUT
* 2 : Thông tin được gửi đi kèm gồm oldPassword,newPassword
* 3 : Thông tin được trả về nếu thay đổi mật khẩu thành công
* 4 : đính jwt vào bearer token để lấy được userid trong jwt.

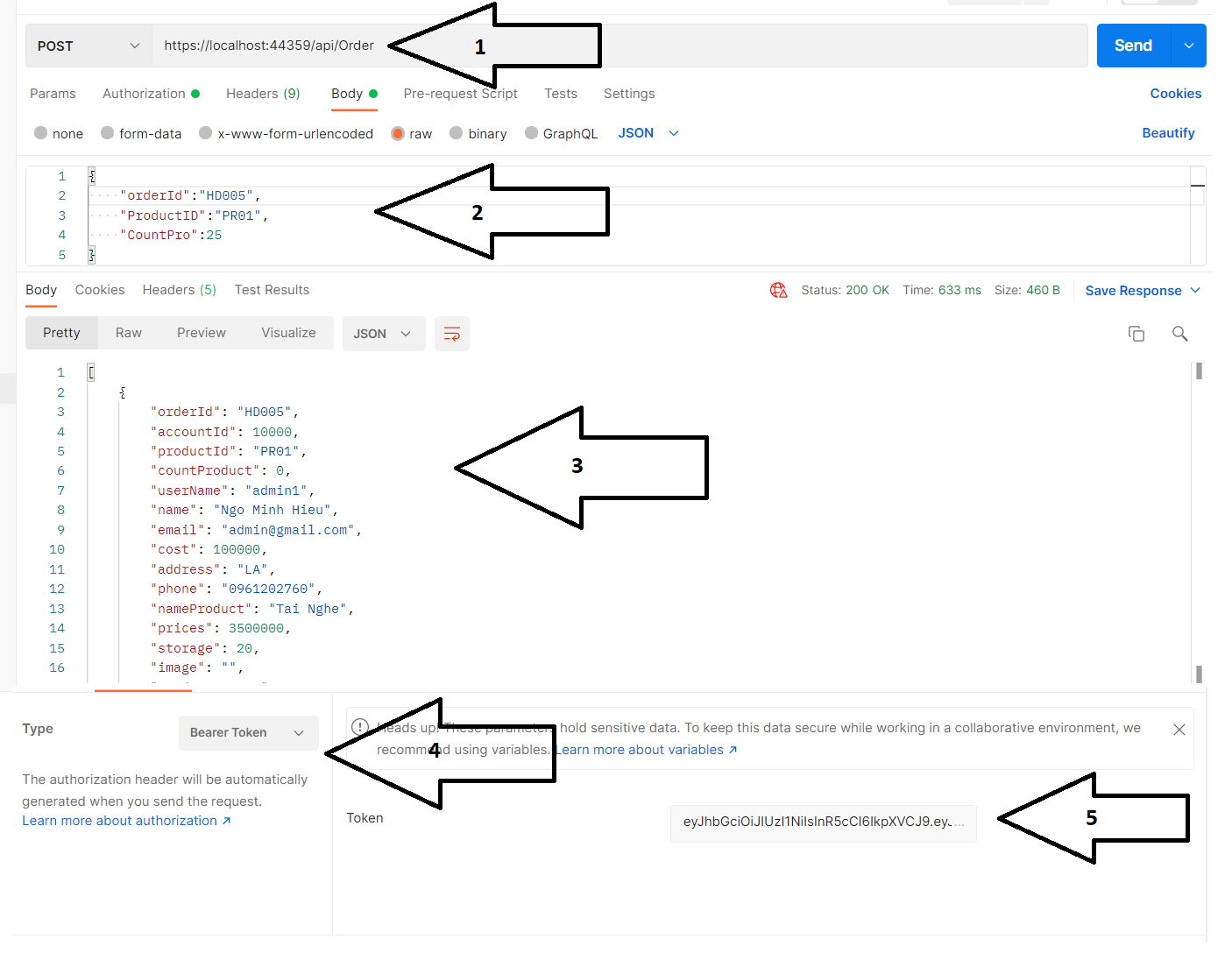
****

Hình 16: API lấy sản phẩm theo ID đơn hàng

* 1 : Đường dẫn tới api : [https://localhost:44359/api/Order/orderID/{idorder}](https://localhost:44359/api/Order/orderID/%7bidorder%7d)
* 1 : Phương thức được sử dụng là phương thức GET, dùng để lấy tất cả sản phẩm trong cùng một order
* 2 : Đính jwt vào bearer token để lấy được userid trong jwt.
* 3 : Thông tin được trả về nếu cập nhật thành công.

****

Hình 17: API lấy tất cả đơn hàng theo ID của user

* 1 : Đường dẫn tới api : <https://localhost:44359/api/Order>
* 1 : Phương thức được sử dụng là phương thức GET, dùng để lấy tất cả order của user
* 2 : Đính jwt vào bearer token để lấy được userid trong jwt.
* 3 : Thông tin được trả về nếu lấy hóa đơn thành công

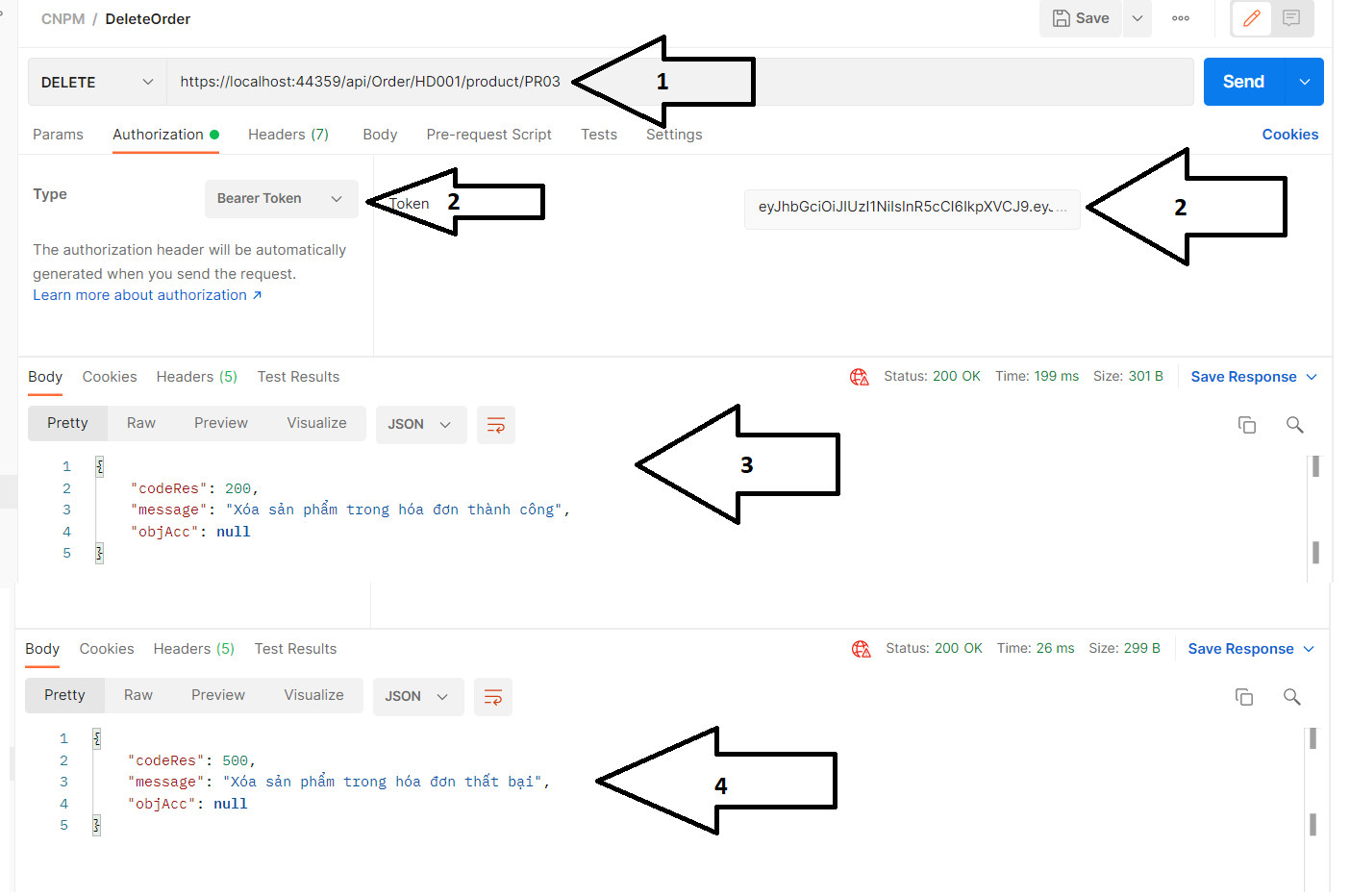
Hình 18: API dùng để thêm sản phẩm vào đơn hàng

* 1 : Đường dẫn tới api : <https://localhost:44359/api/Order>
* 1 : Phương thức được sử dụng là phương thức Post, dùng thêm hóa đơn với sản phẩm
* 2 : Thông tin đính kèm gồm orderID, ProductID, countPro
* 3 : Thông tin được trả về nếu thêm hóa đơn thành công
* 4 : Đính jwt vào bearer token để lấy được userid trong jwt.



Hình 19: API dùng để xóa đơn hàng

* 1 : Đường dẫn tới api : [https://localhost:44359/api/Order/{idOrder}](https://localhost:44359/api/Order/%7bidOrder%7d)
* 1 : Phương thức được sử dụng là phương thức Delete, dùng xóa hóa đơn
* 2 : Thông tin được trả về nếu thêm hóa đơn thành công
* 3 : Đính jwt vào bearer token để lấy được userid trong jwt.



Hình 20: API dùng để xóa sản phẩm trong đơn hàng

* 1 : Đường dẫn tới api :

[https://localhost:44359/api/Order/{idOrder}/product/{idPoduct}](https://localhost:44359/api/Order/%7bidOrder%7d/product/%7bidPoduct%7d)

* 1 : Phương thức được sử dụng là phương thức Delete, dùng xóa sản phẩm trong hóa đơn
* 2 : Đính jwt vào bearer token để lấy được userid trong jwt.
* 3 : Thông tin được trả về nếu thêm hóa đơn thành công
* 4 : Nếu thực hiện thêm 1 lần xóa sản phẩm thì sẽ báo lỗi vì sản phẩm trong order đã được xóa trước đó nên trả về thông báo lỗi

**TÀI LIỆU THAM KHẢO**

1. [**https://www.youtube.com/watch?v=fmvcAzHpsk8&t=1965s**](https://www.youtube.com/watch?v=fmvcAzHpsk8&t=1965s)
2. [**https://docs.microsoft.com/en-us/aspnet/core/tutorials/first-web-api?view=aspnetcore-6.0&tabs=visual-studio**](https://docs.microsoft.com/en-us/aspnet/core/tutorials/first-web-api?view=aspnetcore-6.0&tabs=visual-studio)
3. **Các bài LAB trên lớp thực hành**