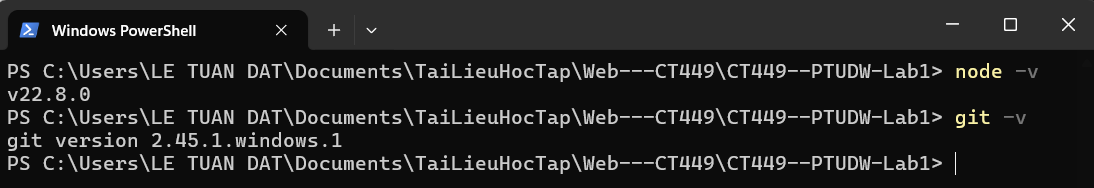
**ỨNG DỤNG CONTACTBOOK - BACKEND - PHẦN 1**

--------

**Bước 0: Cài đặt node và git:**



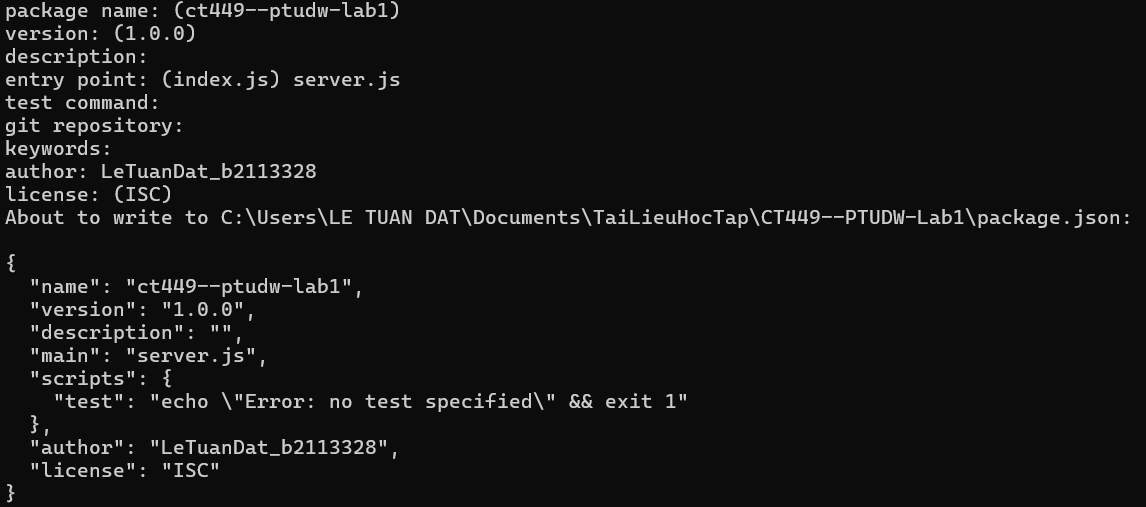
**Bước 1: Tạo ứng dụng Node**

- Tạo thư mục dự án “CT449--PTUDW-Lab1”.

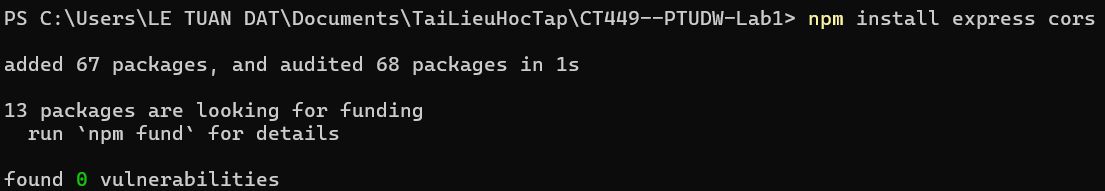
- Khởi tạo Node:



- Thông tin dự án:



- Cài đặt các package cần thiết:

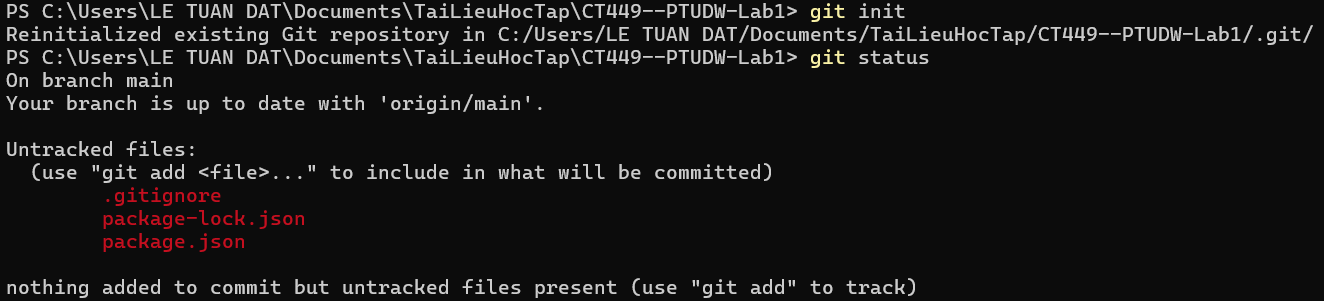


**Bước 2: Quản lý mã nguồn dự án với git và GitHub**

- Tạo tập tin .gitignore: *npx gitignore node*. Tập tin này liệt kê các thư mục, tập tin sẽ không được quản lý bởi git.

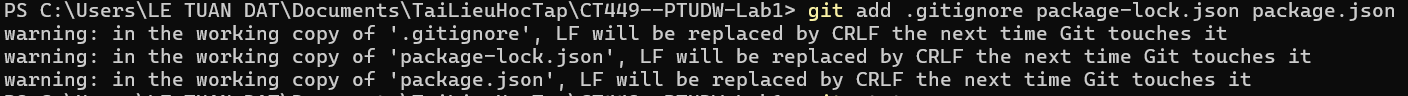


- Trong thư mục gốc dự án, chạy lệnh *git init*. Gõ tiếp *git status*, sẽ xuất hiện 3 tập tin chưa được quản lý bởi git.

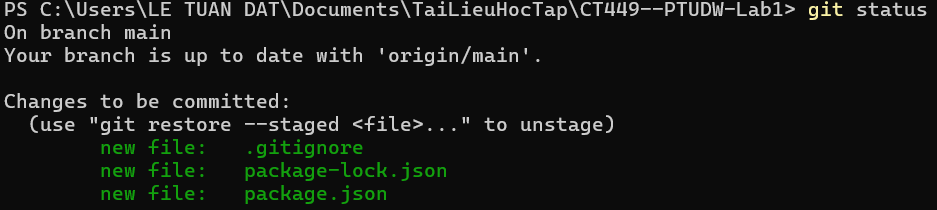


- Cho git quản lý 3 tập tin *.gitignore, package-lock.json, package.json* bằng lệnh:

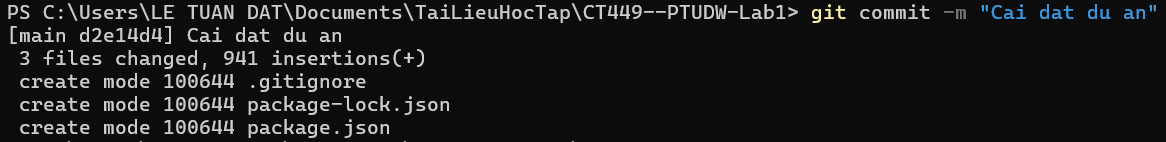
*git add .gitignore package-lock.json package.json*



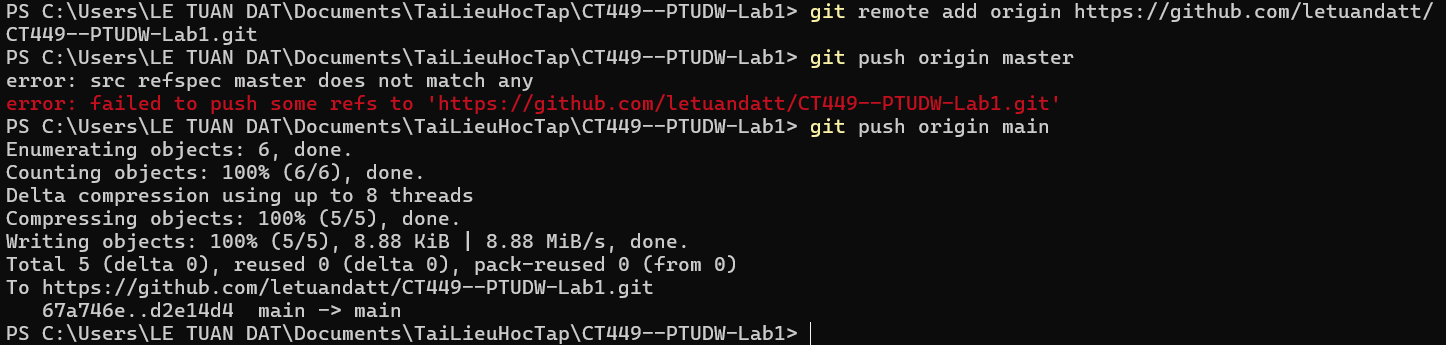
Gõ *git status* để kiểm tra lại:



- Thực thi *git commit -m “Cai dat du an”* để lưu các thay đổi. Tùy chọn *-m* cho phép tạo ghi chú cho commit này.



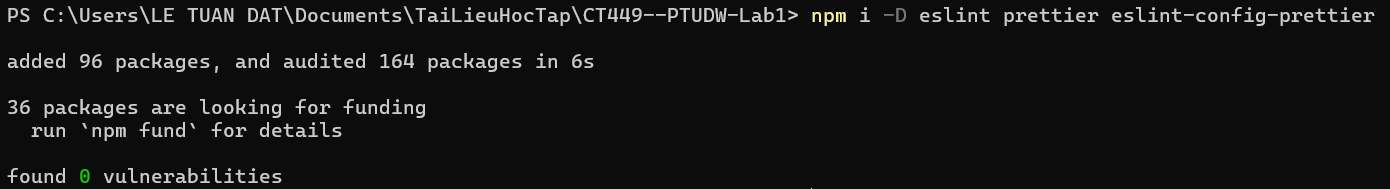
- Upload dự án cục bộ lên GitHub bằng lệnh:



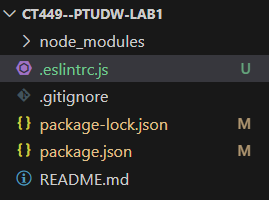


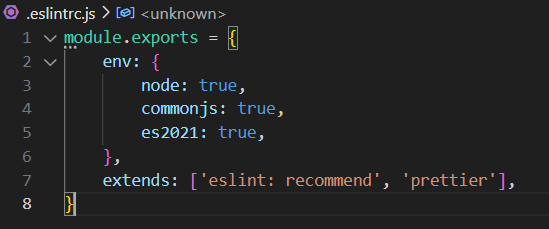
**Bước 3: Cấu hình Visual Studio Code, ESLint và Prettier**

- Cài đặt gói eslint và prettier toàn cục:



- Tạo tập tin *.eslintrc.js* tại thư mục gốc của dự án:



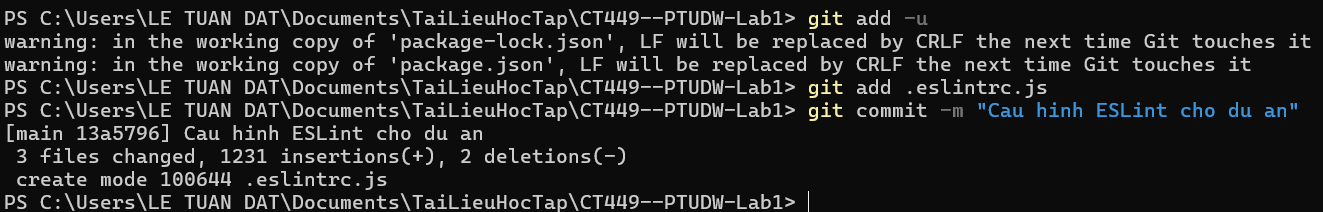


- Lưu thay đổi lên git:

*git add -u*

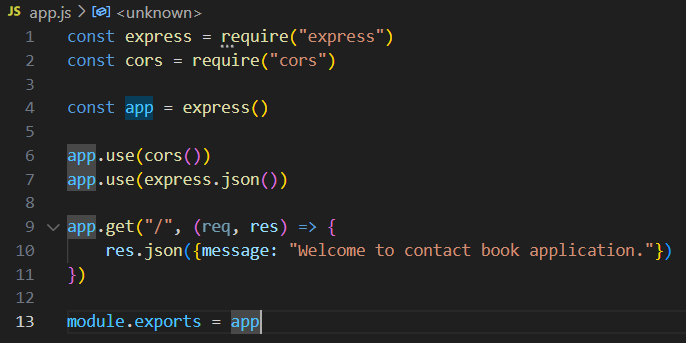
*git add .eslintrc.js*

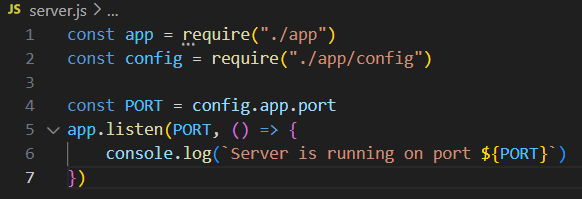
*git commit -m “Cau hinh ESLint cho du an”*



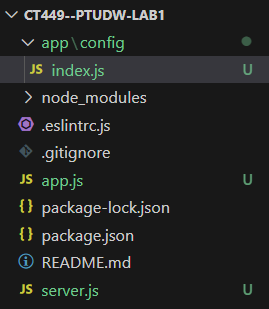
**Bước 4: Cài đặt Express**:

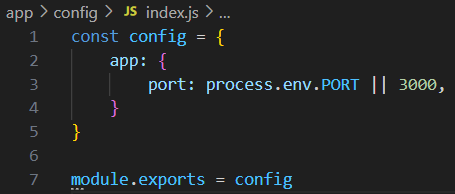
- Tạo lần lượt tập tin app.js và server.js:



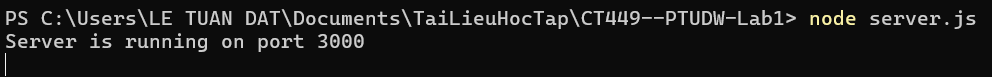


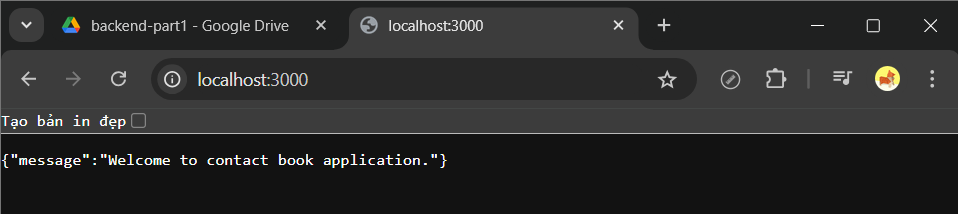
- Lần lượt tạo thư mục app, app/config và tập tin app/config/index.js:



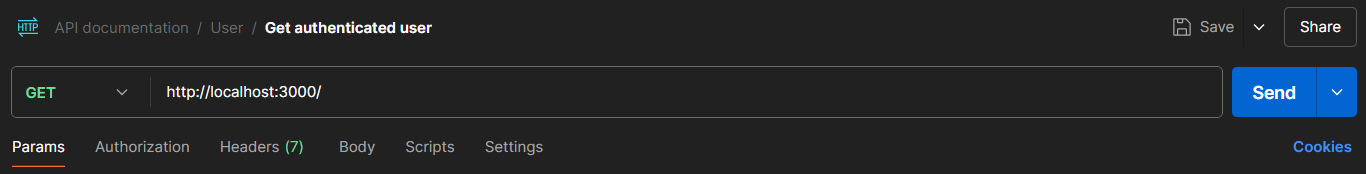


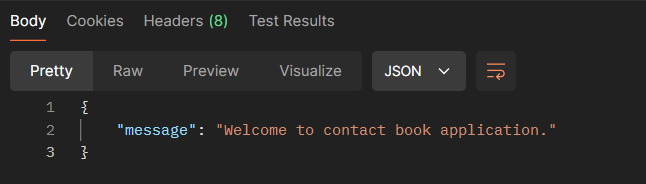
- Mở cửa sổ terminal tại thư mục dự án và thực hiện lệnh: *node server.js*. Truy cập <http://localhost:3000/> để xem kết quả.



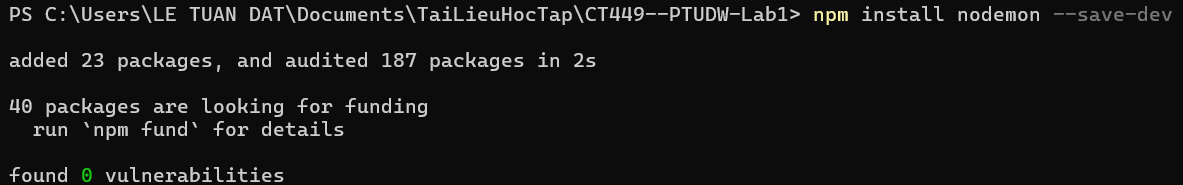


Xem kết quả ở Postman:





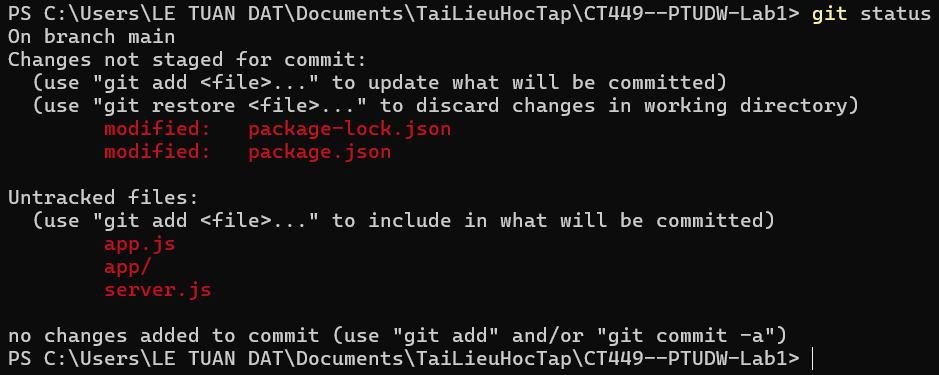
- Cài đặt gói hỗ trợ *nodemon*.

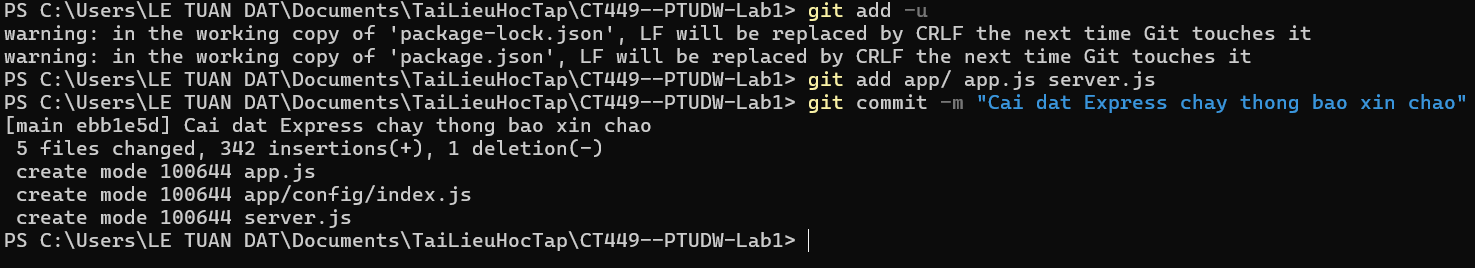


Thay đổi cấu hình mục “scripts” trong tập tin *package.json*:



- Lưu các thay đổi vào git:



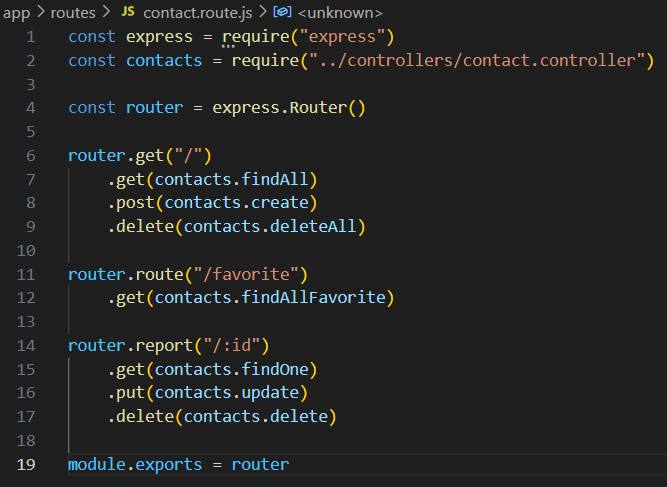


**Bước 5: Định nghĩa Controller và các route**

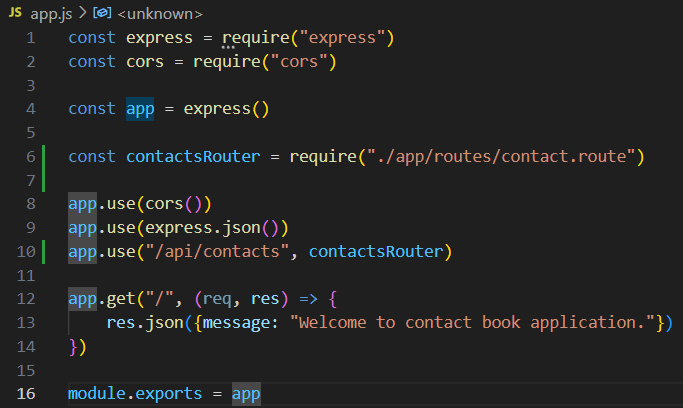
- Tạo thư mục *controllers* trong thư mục *app*. Sau đó, tạo tập tin *contact.controller.js* trong thư mục *controllers* như sau:



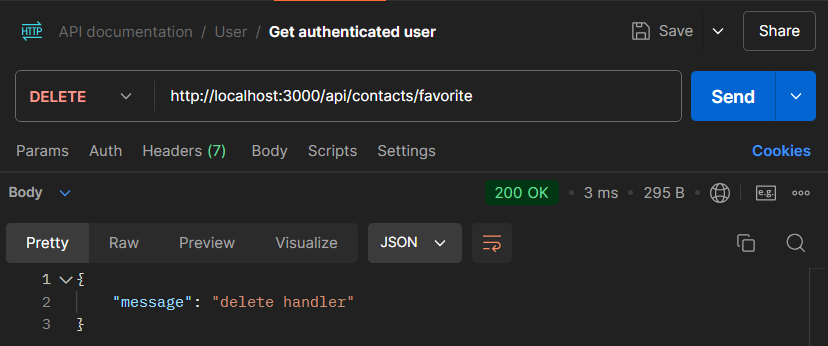
- Tạo thư mục *routes* trong thư mục *app*. Sau đó, tạo tập tin *contact.route.js* trong thư mục *routes* như sau:



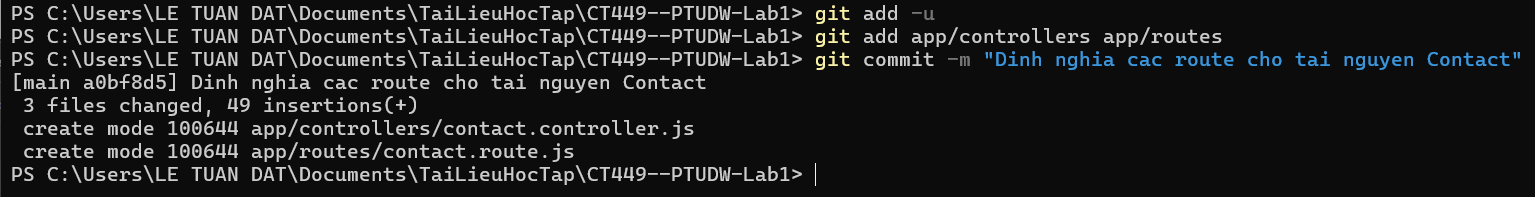
- Hiệu chỉnh tập tin *app.js*:



- Client cần gửi yêu cầu DELETE đến URL: <http://localhost:3000/api/contacts/favorite>

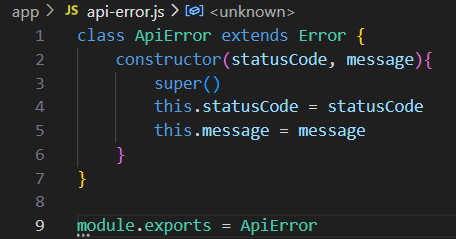


- Lưu các thay đổi lên git:

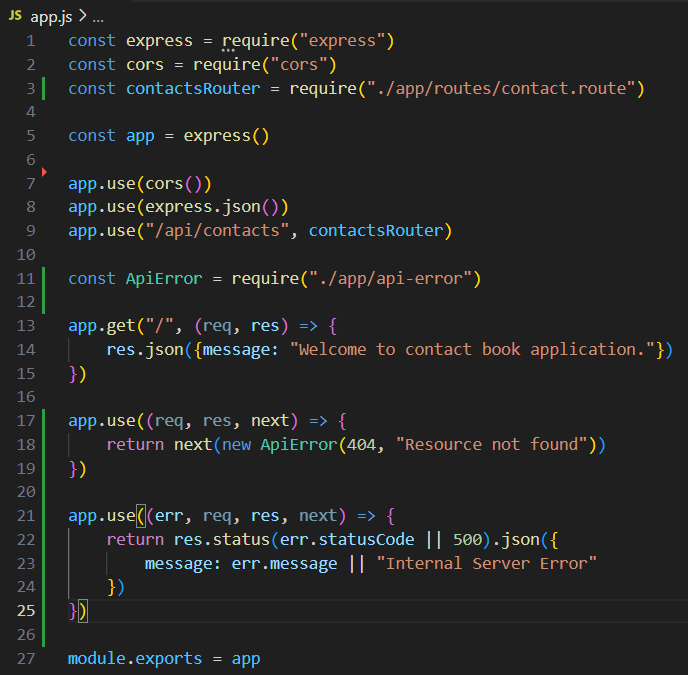


**Bước 6: Cài đặt xử lý lỗi**

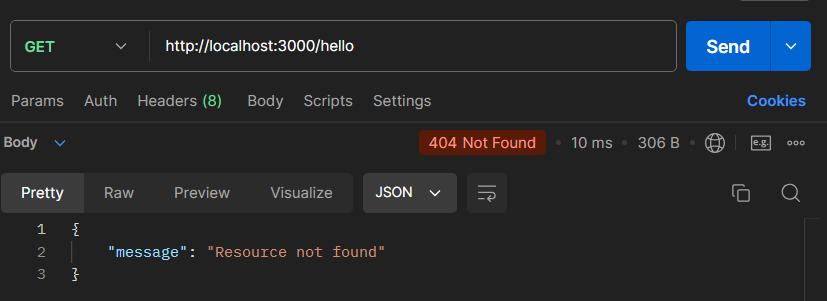
- Tạo tập tin *app/api-error.js*:



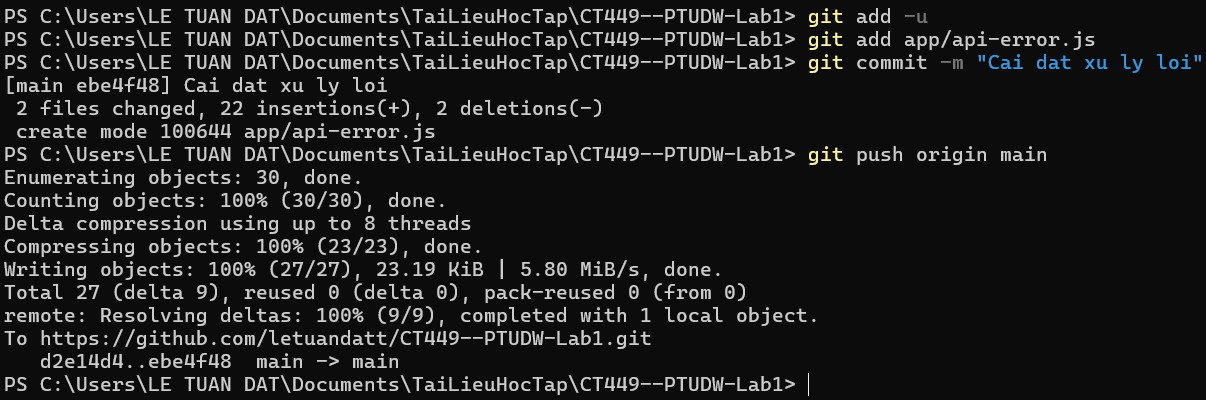
- Mở tập tin *app.js* và thêm vào các middle ware:

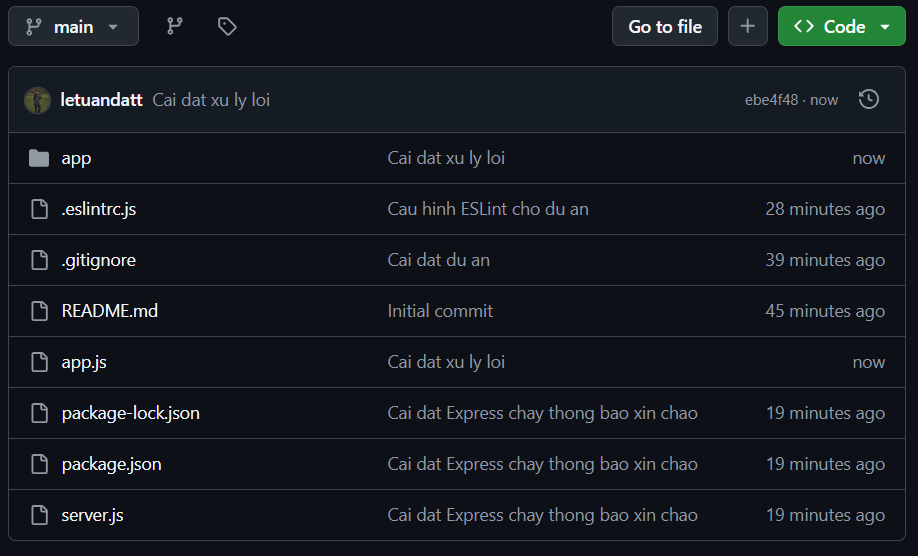


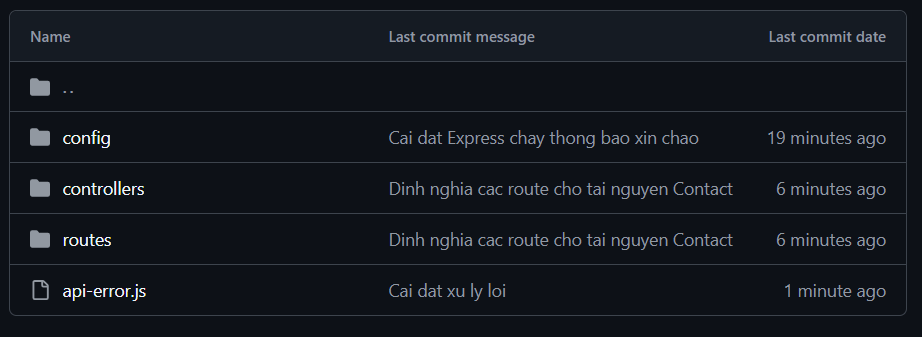
- Sử dụng Postman và gửi yêu cầu đến một URL không được định nghĩa trong ứng dụng, kiểm tra nhận được lỗi 404 và thông báo “Resource not found”.



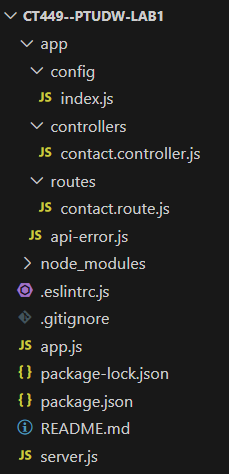
- Lưu các thay đổi vào git và đẩy lên GitHub:







- Cây thư mục dự án đến thời điểm này như sau:



- Đường link GitHub của dự án:

<https://github.com/letuandatt/B2113328_LeTuanDat_BACKEND_1>

**---HẾT---**