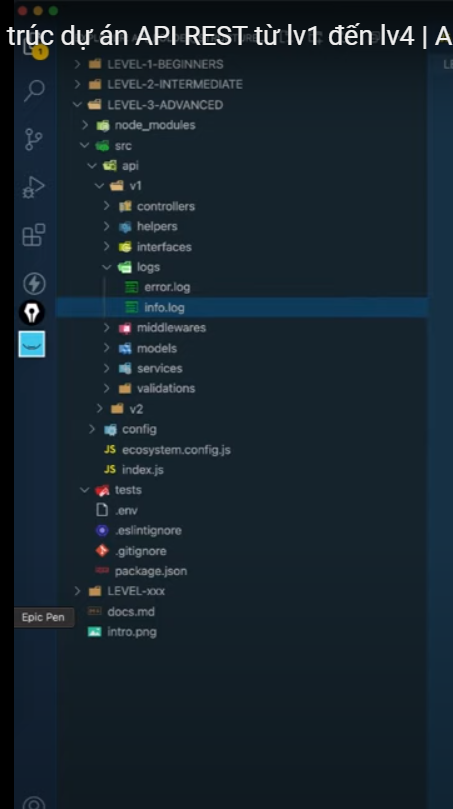
Clean code javascript

<https://github.com/hienvd/clean-code-javascript/>

https://github.com/leonardomso/33-js-concepts

Cấu trúc thư mục:





Note JWT:



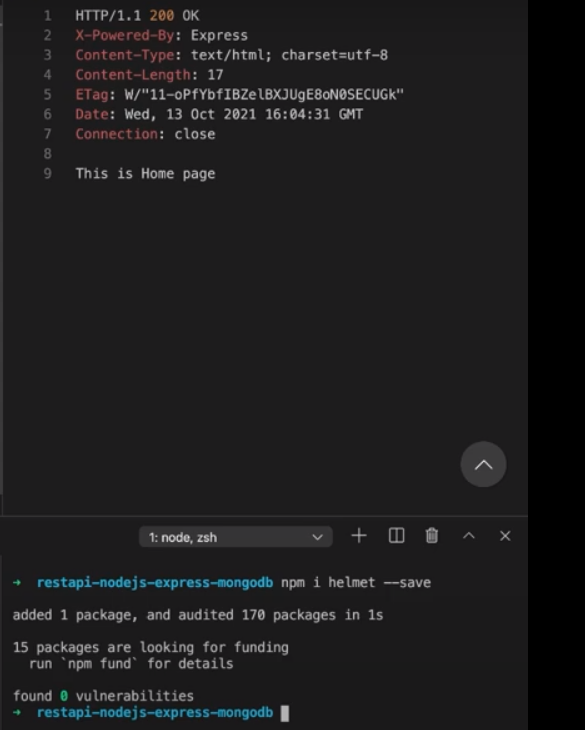


Sử dụng axious.interceptors để kiểm tra trước và sau khi gửi request.

Nếu trước khi gửi request ta check những API cần phải xác thực bằng JWT

. Nếu API đó cần JWT thì ta kiểm tra thời gian hết hạn so với thời gian hiện tại. Nếu AccessToken đã hết hạn ta gọi đến API RefreshToken để nhận lại accessToken mới.

Nhược điểm: Thời gian trong máy mỗi user sẽ khác nhau -> Hacker có thể tận dụng thay đổi thời gian để lạm quyền.



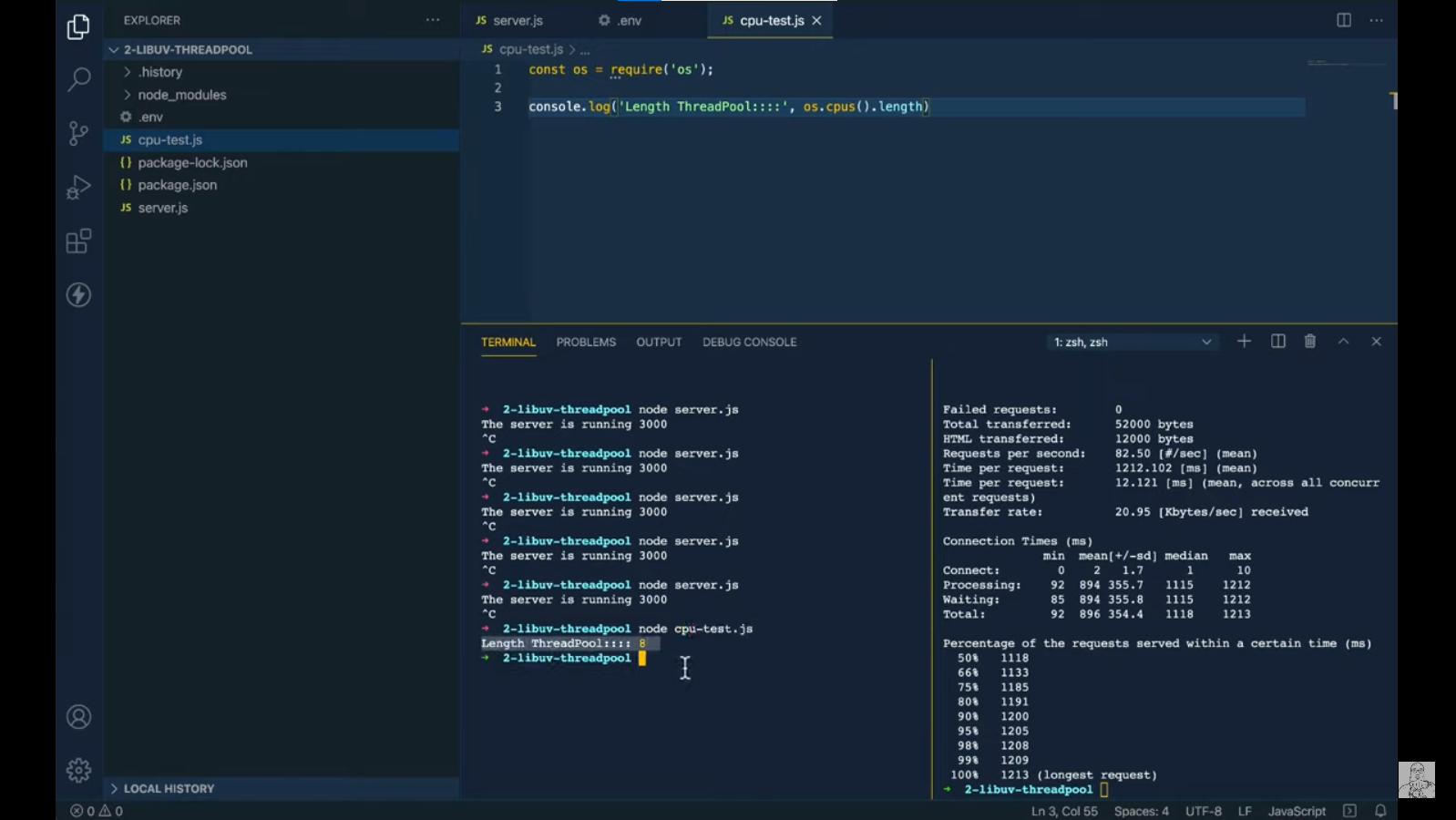
Dùng package helmet để tránh khi gọi API server trả về loại công nghệ dùng để xây dựng API => Hạn chế nguy cơ bị tấn công bởi hacker do các lỗ hổng bảo mật chưa được vá => (Middleware Helmet rất quan trọng trong quá trình xây dựng REST API)

Debounce => Ứng dụng closure để check nếu trước đó đã có sự kiện trong setTimeOut nếu người dùng kích hoạt sự kiện thì hủy sự kiện đang có trong setTimeOut và setTimeout mới reset lại biến delay

Throttle

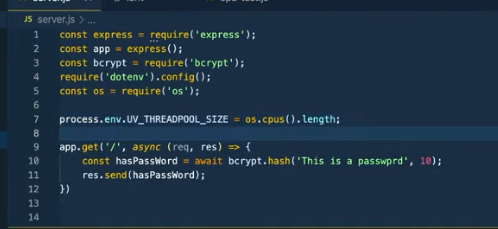
LRUCache => Cache lại những dữ liệu được truy cập nhiều





Tăng tốc ứng dụng NodeJS bằng cách tận dụng tất cả các core và thread =>

Code trên check xem máy có bao nhiêu luồng



Trong thực tế không nên dùng như trên vì cần nhường một số luồng cho các hoạt động khác.

