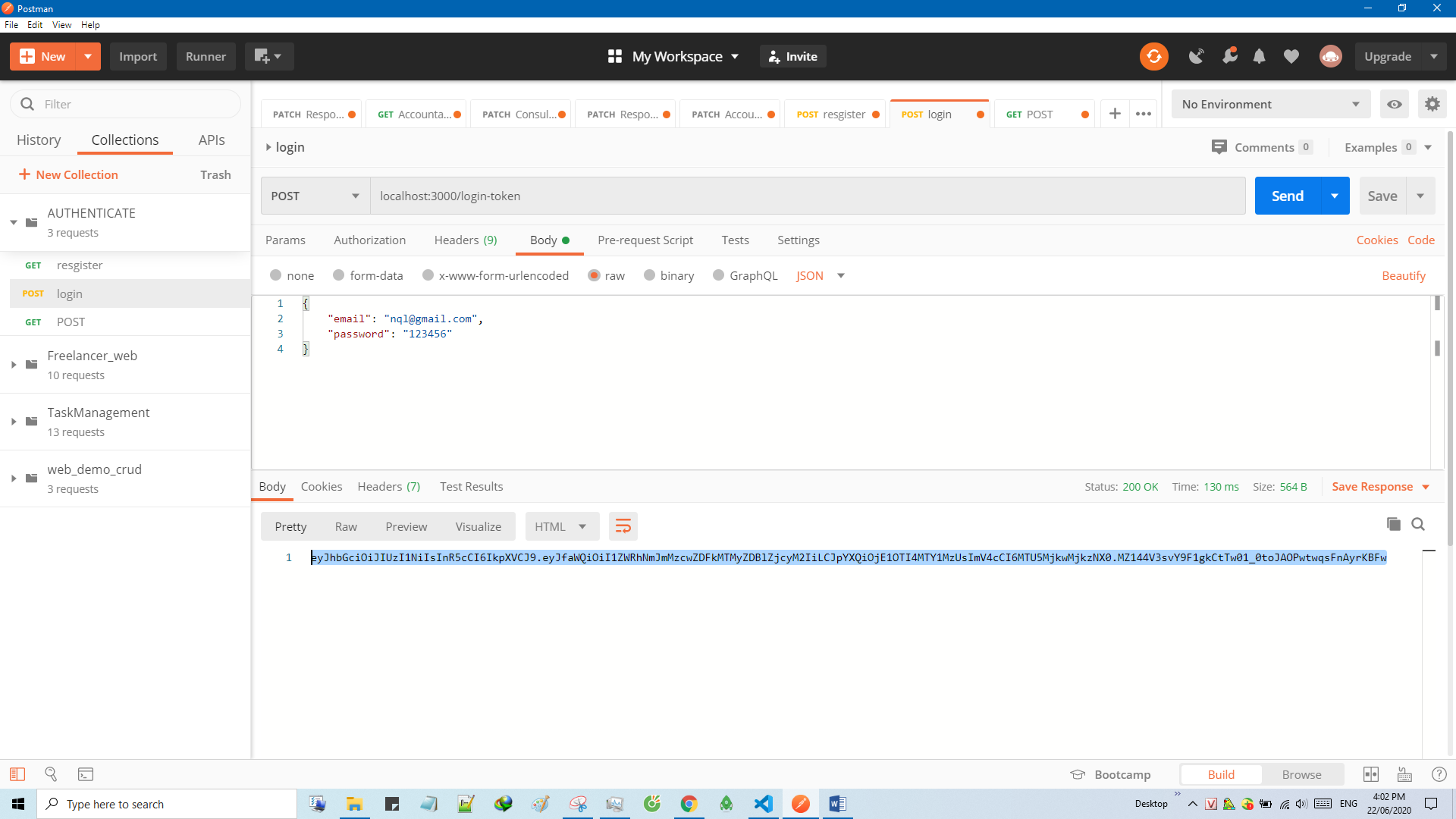
**BÁO CÁO DEMO AUTHENTICATION**

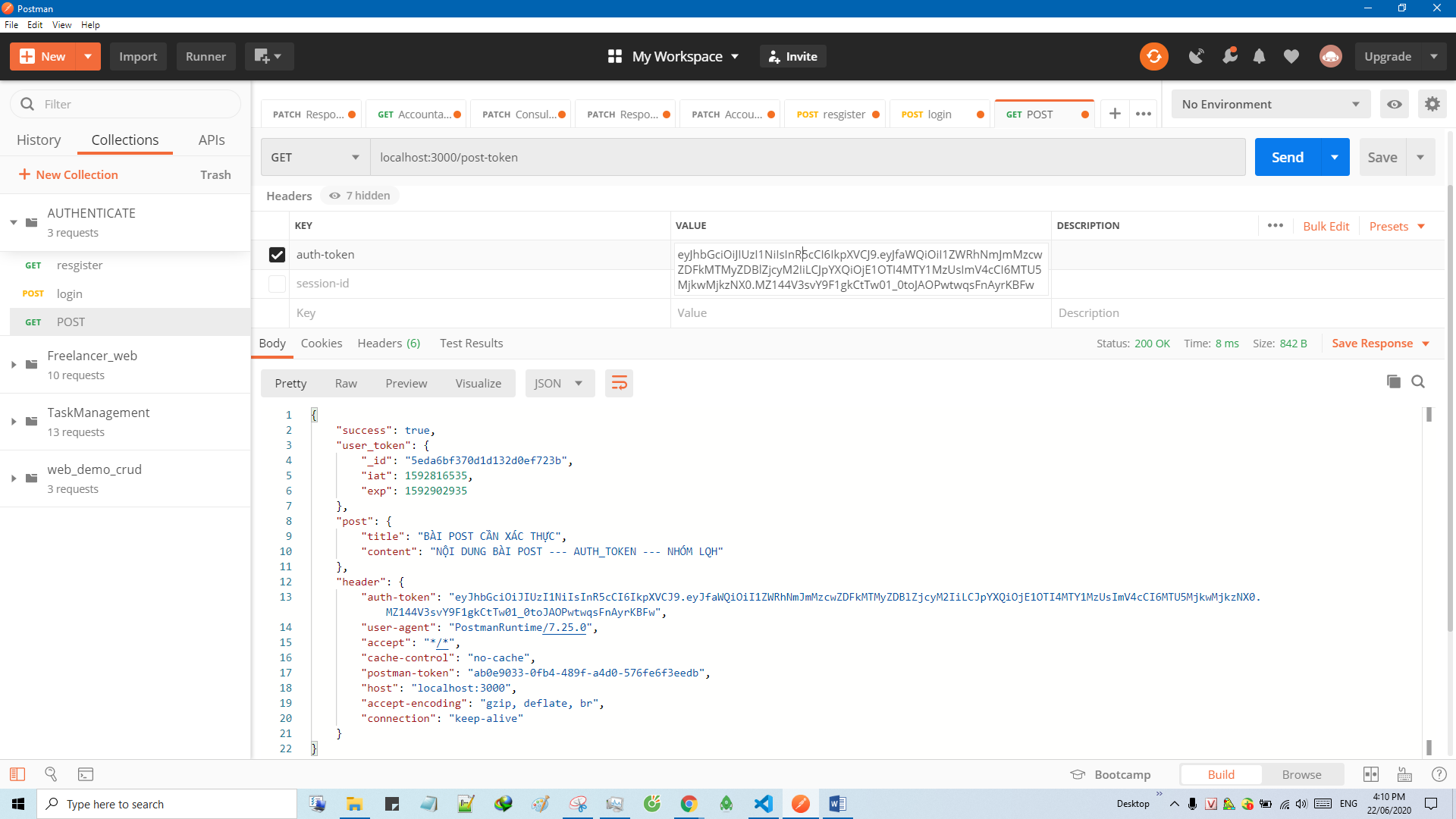
Với đề tài bảo mật web service, nhóm chúng em sẽ làm bảo mật về xác thực, sử dụng authorization, auth-token, và session-id. Do thời gian hạn chế nên bọn em không làm giao diện, mà sẽ test với phần mềm postman. Dưới đây là hình ảnh về phần demo của nhóm em.

Với auth-token thì chúng em sẽ cung cấp cho người dùng khi đăng nhập một chuỗi token, được mã hóa từ id của người dùng với, một chuỗi TOKEN\_SECRET đặt sẵn (auth-token), và giới hạn thời gian tồn tại của token này là 1 ngày. Khi người dùng đăng nhập, server tạo chuỗi token xong sẽ đẩy lên header và khi đó người dùng muốn thực hiện một request gì đó, server sẽ kiểm tra auth-token của người này có giống với người đăng nhập hay không và cho phép thực hiện request.

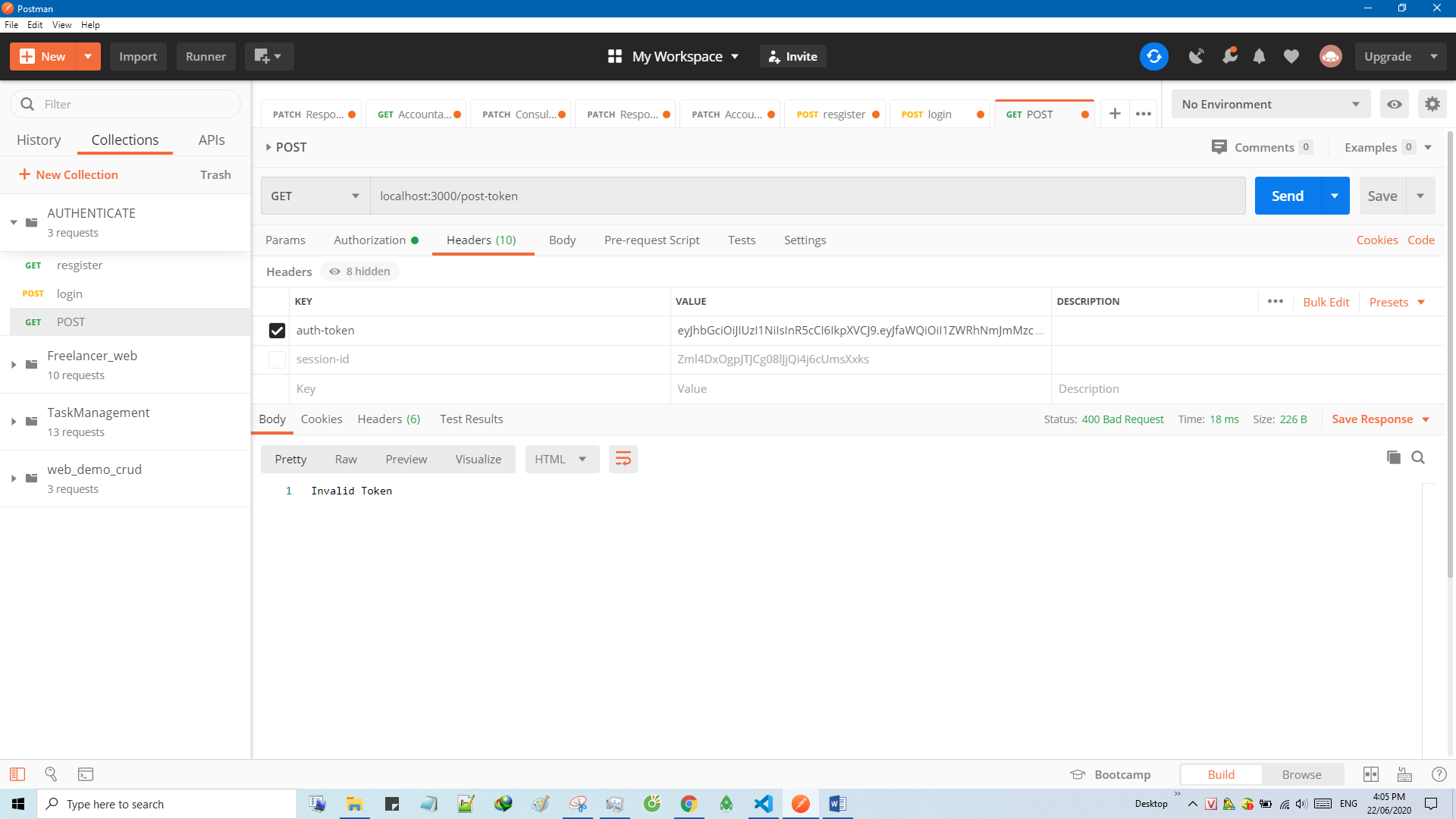


Khi login thành công thì nhóm em sẽ cho server trả về chuỗi token của người này, để lấy chuỗi token đó để test. Khi người dùng gửi request đi. Nếu token của người này đúng thì server sẽ trả về status 200 và kết quả, còn sai hoặc không có thì sẽ trả về status 400 với thông báo lỗi.

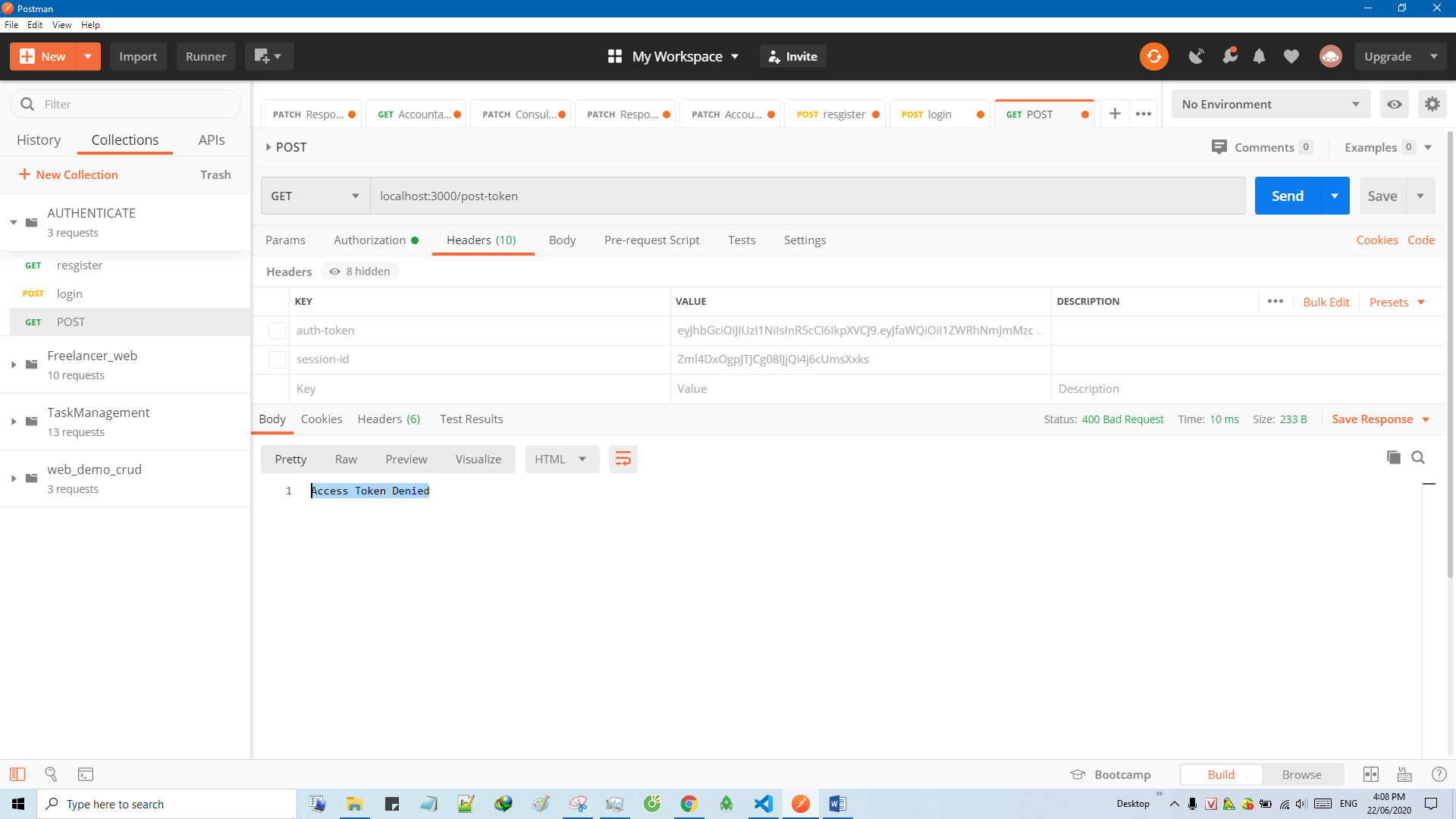
Nếu đúng:



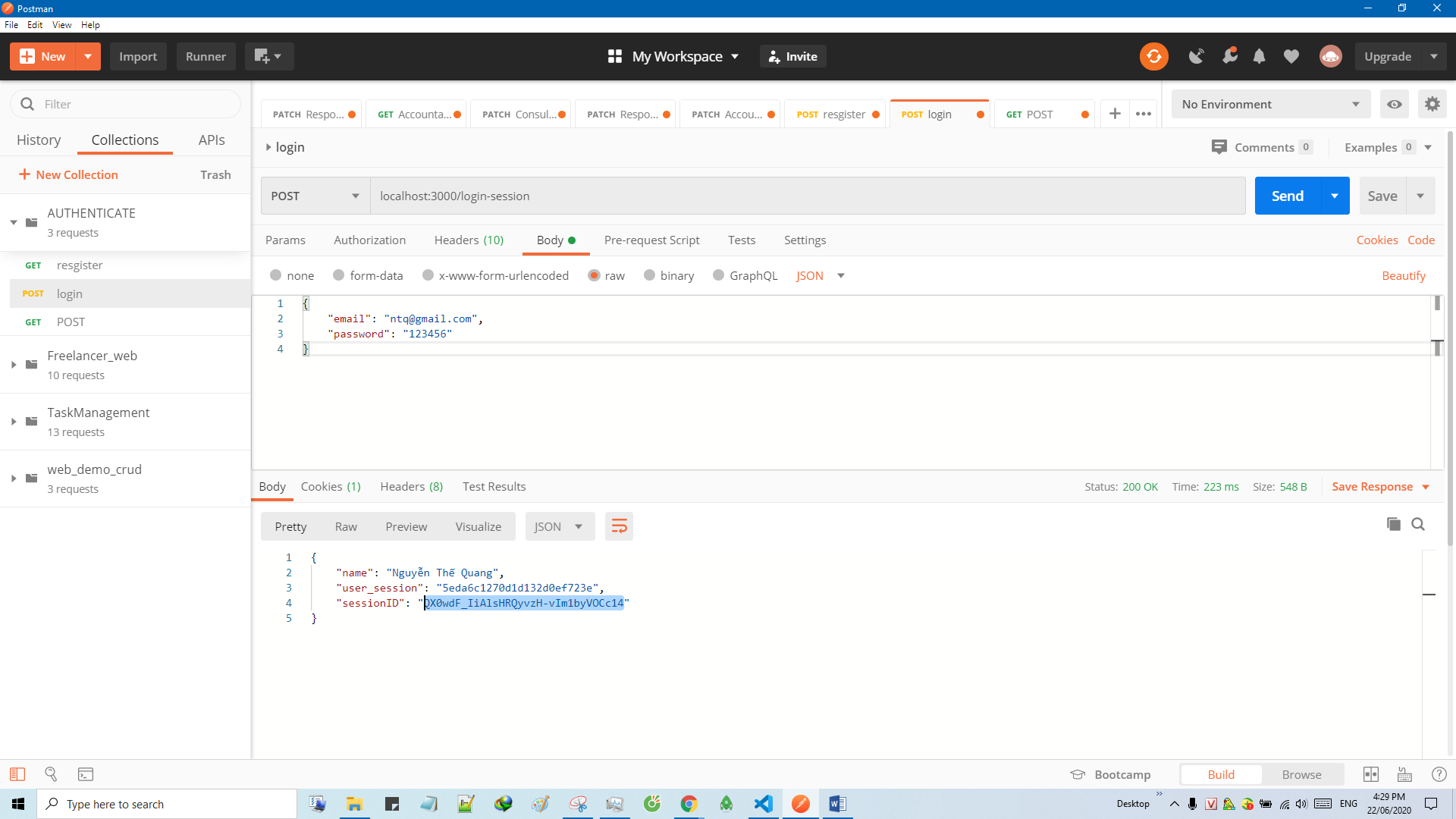
Nếu token sai (một người khác người đăng nhập):



Nếu không có token, không đẩy lên header (tức là chưa đăng nhập):

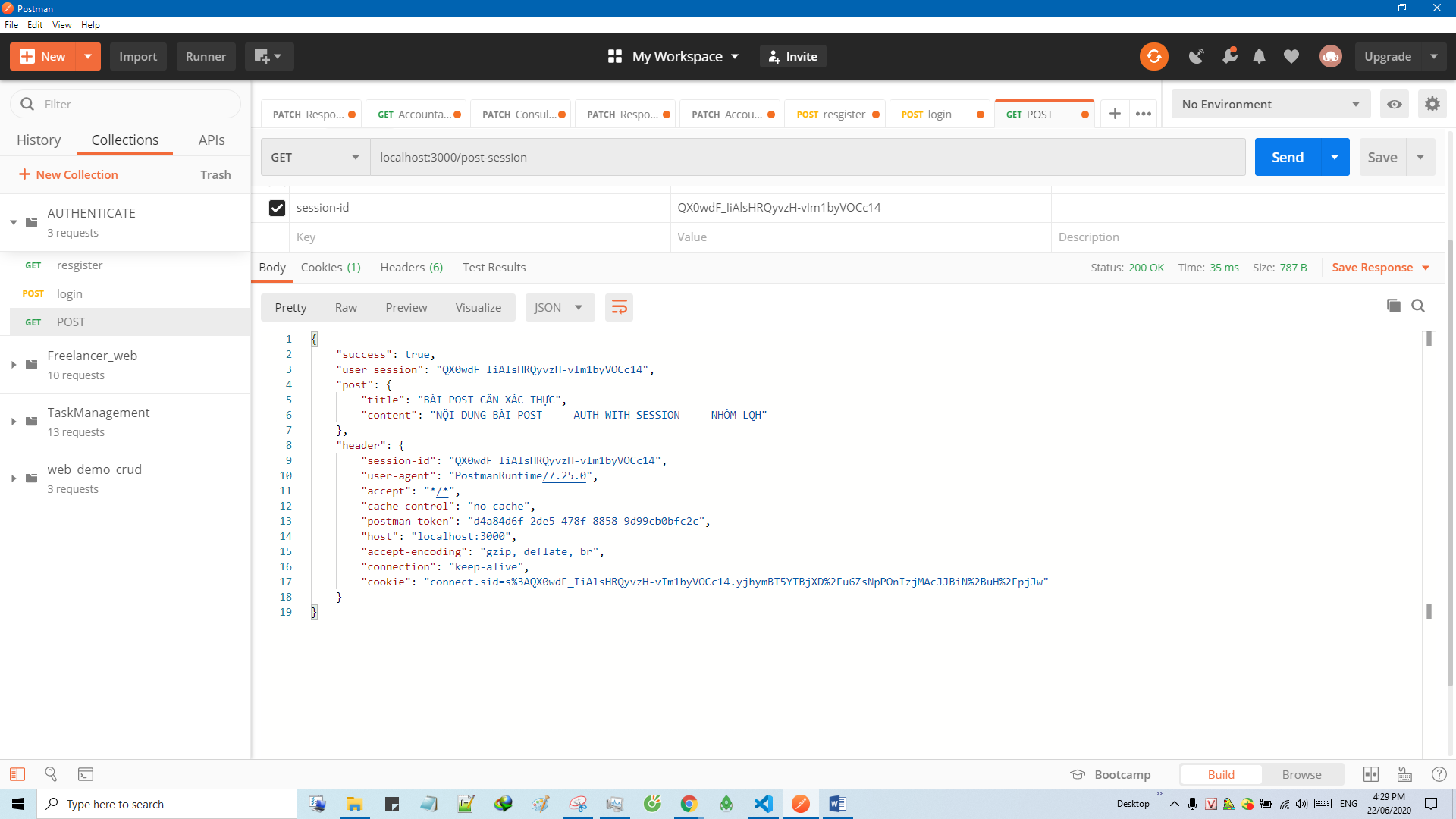


Với xác thực dùng session-id thì chúng em sẽ cung cấp cho người dùng khi đăng nhập một session, được cấu hình cài đặt với các thông số secret, resave, store, expires,… và giới hạn thời gian tồn tại của session này cũng là 1 ngày. Khi người dùng đăng nhập, server tạo chuỗi một session xong, sau đó sẽ đẩy sesion-id lên header và khi đó người dùng muốn thực hiện một request gì đó, server sẽ kiểm tra session-id của người này có giống với người đăng nhập hay không và cho phép thực hiện request.

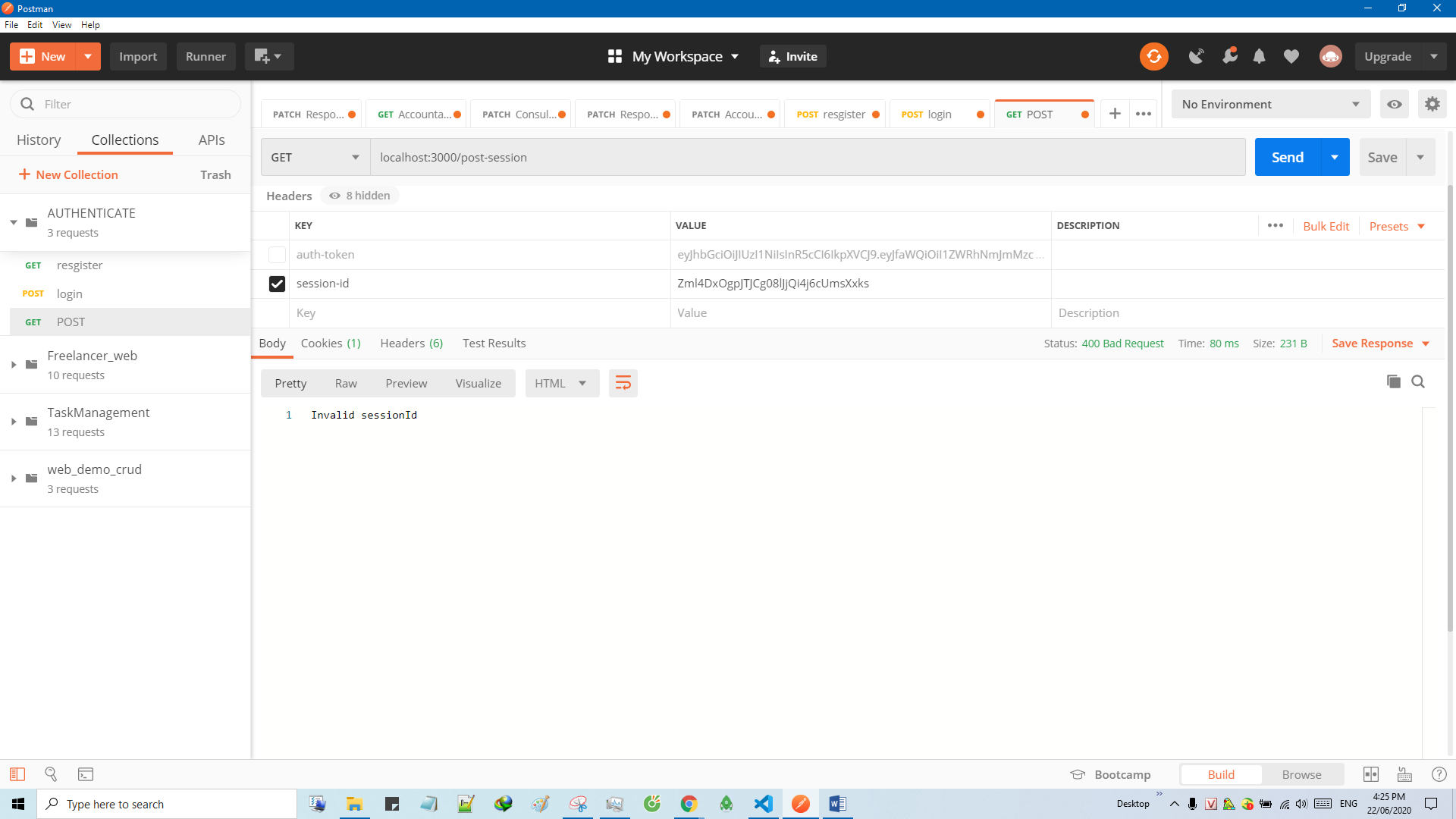


Khi đăng nhập thành công thì server sẽ trả về status 200, và thông điệp như trên. Khi người dùng gửi request đi. Nếu session-id của người này đúng với trên header thì server sẽ trả về status 200 và kết quả, còn sai hoặc không có thì sẽ trả về status 400 với thông báo lỗi.

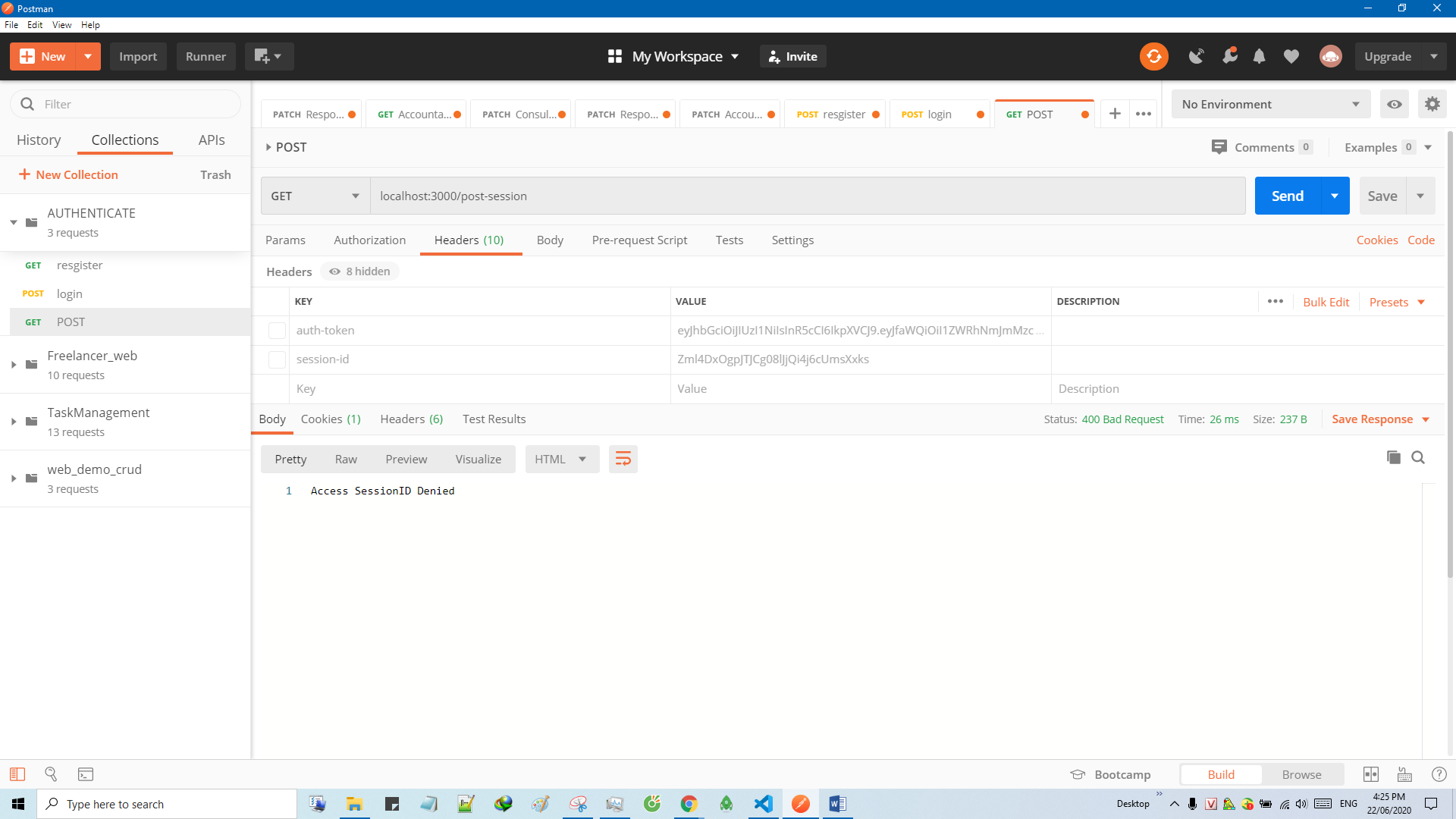
Nếu đúng:



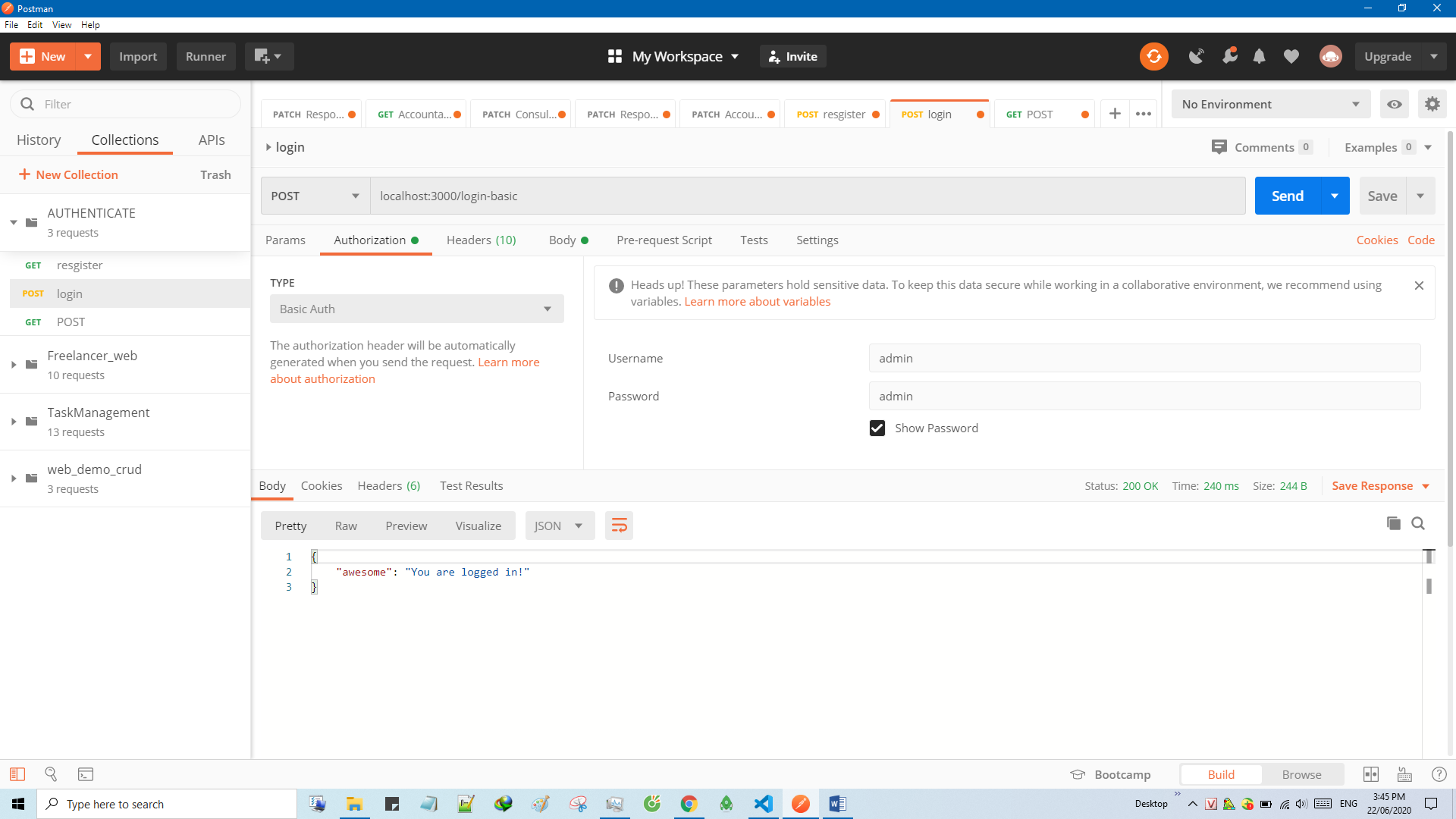
Nếu session-id sai (người khác đăng nhập):

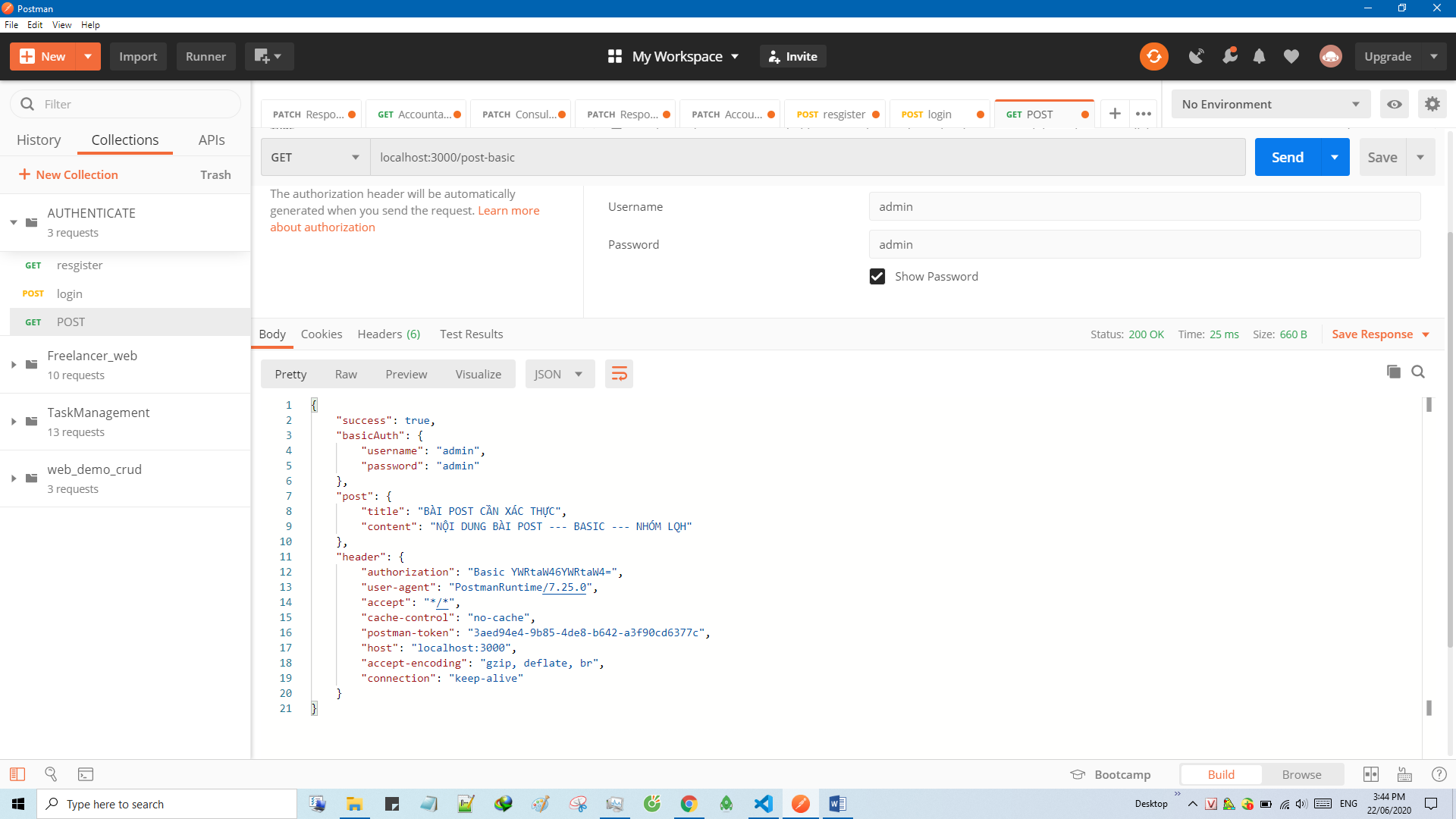


Nếu session-id không có (chưa đăng nhập):



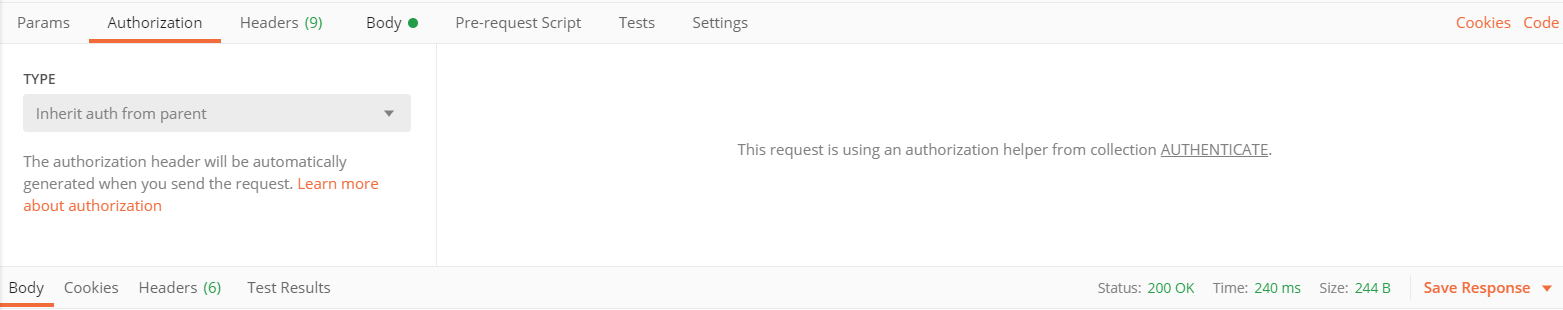
Với authorization bọn em sẽ sử dụng Basic Auth để cung cấp một username và password mặc định (username: admin – password: admin), sau đó convert sang một mã base64 và lưu vào phần authorization trên header. Khi người dùng muốn thực hiện một thao tác nào đó (ví dụ như GET một bài post nào đó), thì server sẽ kiểm tra phần authorization trên header của request, convert lại sang chuỗi ascii rồi lấy được username và password của authorization của người đăng nhập này, kiểm tra so sánh với username và password mặc định đặt ra. Nếu đúng sẽ trả về status 200 và kết quả, còn sai thì sẽ trả về status 500 nếu không có basicAuth, còn nếu sai username, password thì sẽ trả về status 400 và message là “Access authorization denied ”

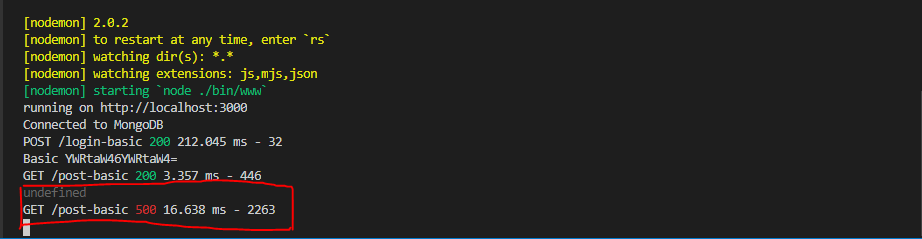




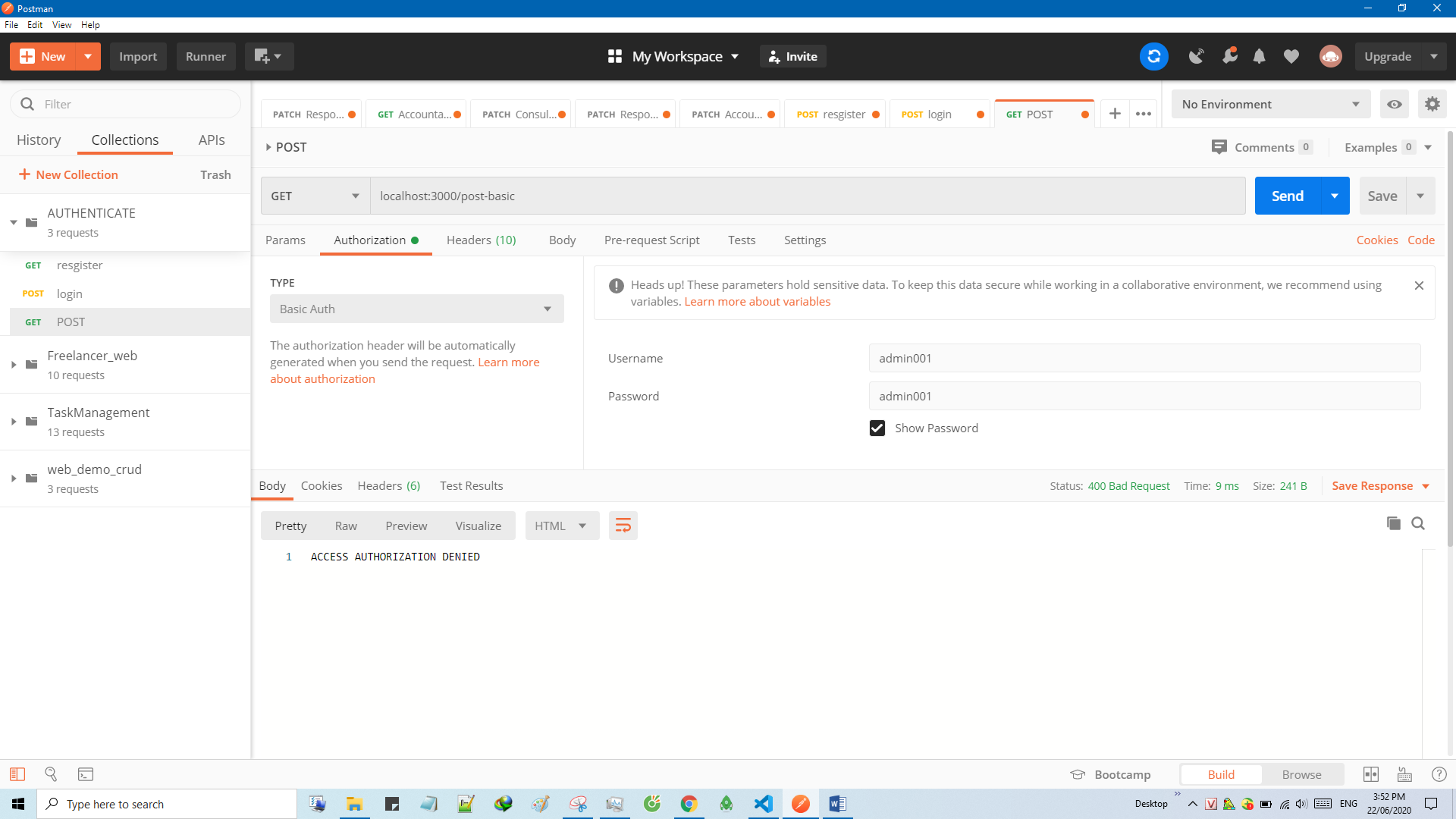
Đây là khi người dùng có sử dụng đúng basicAuth authorization

Còn với trường hợp người dùng không được cung cấp một BasicAuth để dùng cho Authorization thì khi gửi request sẽ báo lỗi





Còn nếu mà user name hoặc password của người dùng sai thì nó sẽ trả về như sau. Ở đây chúng em sẽ đổi user name và password thành admin001.



Trên đây là demo xác thực của chúng em, do để làm rõ, phân biệt từng cách xác thực nên chúng em tách riêng ra từng phần. Tuy nhiên chúng ta cũng có thể gộp chung và sử dụng cả ba cách vào một để tăng tính bảo mật cho website.