

BÀI TẬP TUẦN 4

Nhắc lại

Đã nắm được tất cả các API đăng bài và giao diện đi kèm

Tiếp tục với nhóm API sửa/xóa/báo cáo bài và các giao diện

Tiếp tục với nhóm API và giao diện liên quan đến bình luận

Danh sách API

Danh sách 41 API cần được phát triển:

Login, logout, signup, get_verify_code, check_verify_code, change_info_after_signup,
get_list_posts, get_post, add_post, edit_post, delete_post, get_comment, set_comment,
report_post, like, search, get_saved_search, del_saved_search, get_user_friends,
get_user_info, set_user_info, get_list_videos, get_list_blocks, set_block,
set_accept_friend, get_requested_friends, set_request_friend, get_push_settings,
set_push_settings, change_password, check_new_version, set_devtoken, get_conversation,
delete_message, get_list_conversation, delete_conversation, get_list_suggested_friends,
check_new_item, get_notification, set_read_message, set_read_notification,

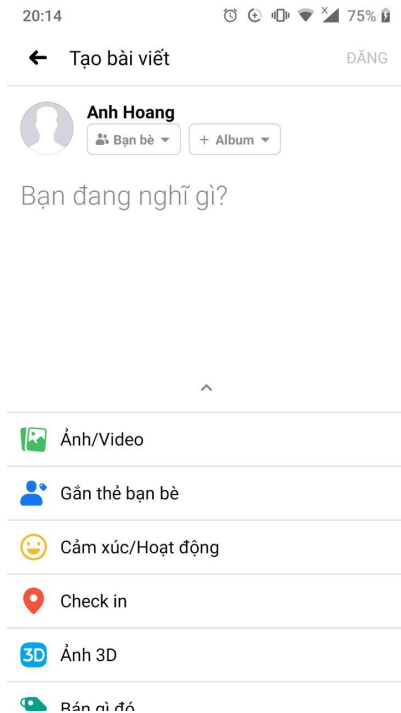
Mục lục

1. Giao diện sửa bài
2. API xóa bài
3. Giao diện xóa bài
4. API báo cáo bài
5. Giao diện báo cáo bài
6. API thích bài
7. API xem bình luận bài
8. API đăng bình luận bài
9. Giao diện xem bình luận

Mục lục

1. **Giao diện sửa bài**
2. API xóa bài
3. Giao diện xóa bài
4. API báo cáo bài
5. Giao diện báo cáo bài
6. API thích bài
7. API xem bình luận bài
8. API đăng bình luận bài
9. Giao diện xem bình luận

Giao diện sửa bài



Nhìn chung giống với giao diện đăng bài, chẳng hạn:

Chỉ có một chế độ hiển thị công khai, không tùy chọn Album

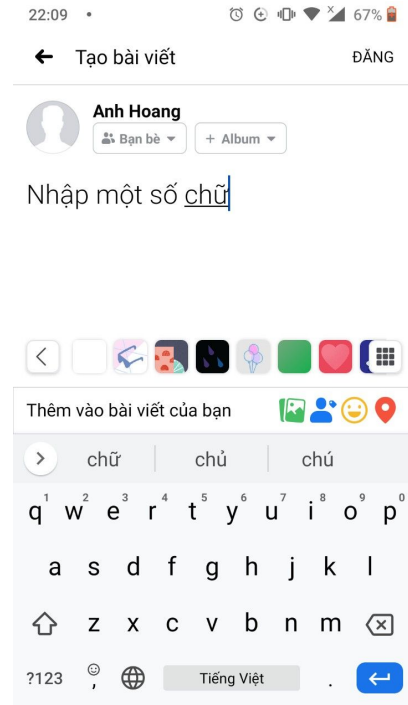
Không có gắn thẻ bạn bè

Danh sách các Cảm xúc/Hoạt động được định nghĩa sẵn và lưu trong ứng dụng

Không có chức năng check in, ảnh 3D...

Không tùy chọn ảnh/màu nền của bài viết

Thay đổi giao diện khi bắt đầu nhập một số nội dung

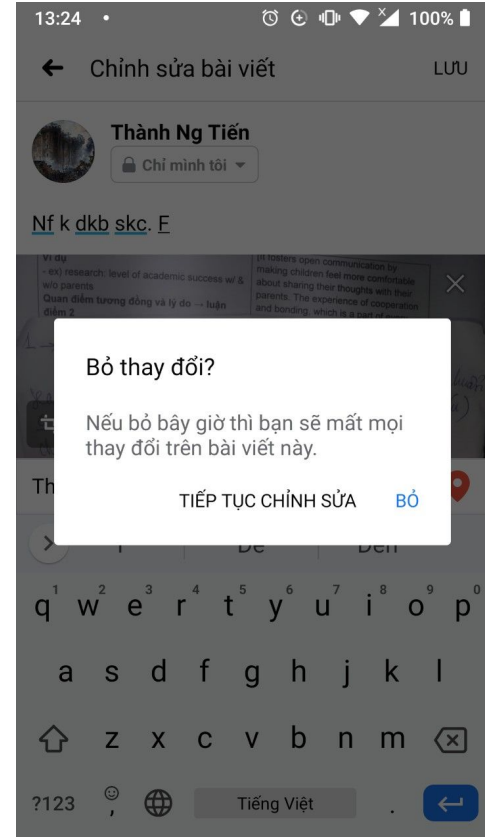


Giao diện sửa bài (2)

Điểm khác biệt lớn nhất là giao diện khi người dùng đang sửa bài thì nhấn nút back trên thiết bị hoặc ở menu.

Điểm khác biệt nữa là trước khi hiện lên giao diện sửa bài, người lập trình viên phải có các xử lý để gán nội dung cho các đối tượng giao diện (vùng văn bản, nút nhấn) cũng như chuyển đổi từ emoticons sang dạng văn bản.

Các link liên kết cũng được chuyển đổi về dạng văn bản.



Mục lục

1. Giao diện sửa bài
2. **API xóa bài**
3. Giao diện xóa bài
4. API báo cáo bài
5. Giao diện báo cáo bài
6. API thích bài
7. API xem bình luận bài
8. API đăng bình luận bài
9. Giao diện xem bình luận

delete_post

Input

NO	Tên parameter	Type	NN	Mô tả
1	token	string	0	
2	id	integer	0	id bài viết

Output

NO	Tên parameter	Type	NN	Mô tả
1	code	string	0	(tham khảo phần response common)
2	message	string	0	

Mô tả delete_post

API thực hiện việc xóa một bài viết của một tài khoản của người dùng

Request dạng POST

Tham số: **token, id** (của bài viết)

Kết quả đầu ra: Nếu thành công thì mã thông báo thành công được trả về, bài viết bị xóa.
Nếu không thành công thì sẽ có các thông báo lỗi tương ứng

Các test case cho delete_post

1. Người dùng truyền đúng mã phiên đăng nhập, id bài viết

Kết quả mong đợi: 1000 | OK (Thông báo thành công), gửi cho ứng dụng các thông tin cần thiết.

2. Người dùng gửi sai mã phiên đăng nhập (mã bị trống hoặc quá ngắn hoặc mã phiên đăng nhập cũ).

Kết quả mong đợi: ứng dụng sẽ phải đẩy người dùng sang trang đăng nhập. Xem lại test case 3 của change_info_after_signup

Các test case cho delete_post (2)

3. Người dùng truyền đúng mã phiên đăng nhập, id bài viết. Nhưng bài viết đã bị khóa (do vi phạm tiêu chuẩn cộng đồng hoặc bị hạn chế tại quốc gia)

Kết quả mong đợi: mã lỗi 9992 và bài viết bị biến mất trong trang hiện tại. Nếu là trang chủ thì ứng dụng sẽ xóa bài viết đó. Nếu là trang cá nhân thì có thể xóa bài viết đó hoặc làm mới lại trang cá nhân (tùy thuộc tình huống).

4. Người dùng truyền đúng mã phiên đăng nhập, id bài viết. Nhưng người dùng đã bị khóa tài khoản (do hệ thống khóa đi).

Kết quả mong đợi: ứng dụng sẽ phải đẩy người dùng sang trang đăng nhập. Xem lại test case 3 của change_info_after_signup

Các test case cho delete_post (3)

5. Người dùng truyền đúng mã phiên đăng nhập, id bài viết. Nhưng hệ thống không thể xóa bài viết đi được

Kết quả mong đợi: mã lỗi báo hệ thống không thể xóa được do lỗi DB. Nhưng ở phía ứng dụng, cần hiện thông báo khác, chẳng hạn “Không thể kết nối Internet”

6. Người dùng truyền đúng mã phiên đăng nhập, nhưng sai id bài viết.

Kết quả mong đợi: mã lỗi trả về báo giá trị tham số trả về bị sai. Nhưng tất nhiên ứng dụng sẽ phải có thông báo khác đi

Các test case cho delete_post (4)

7. Người dùng truyền đúng mã phiên đăng nhập, id bài viết. Nhưng đang kết nối thì mạng Internet bị ngắt

Kết quả mong đợi: ứng dụng cần hiện thông báo “Không thể kết nối Internet” càng sớm càng tốt

CÂU HỎI 1: THEO EM CÒN CÓ CÁC TEST CASE NÀO NỮA?

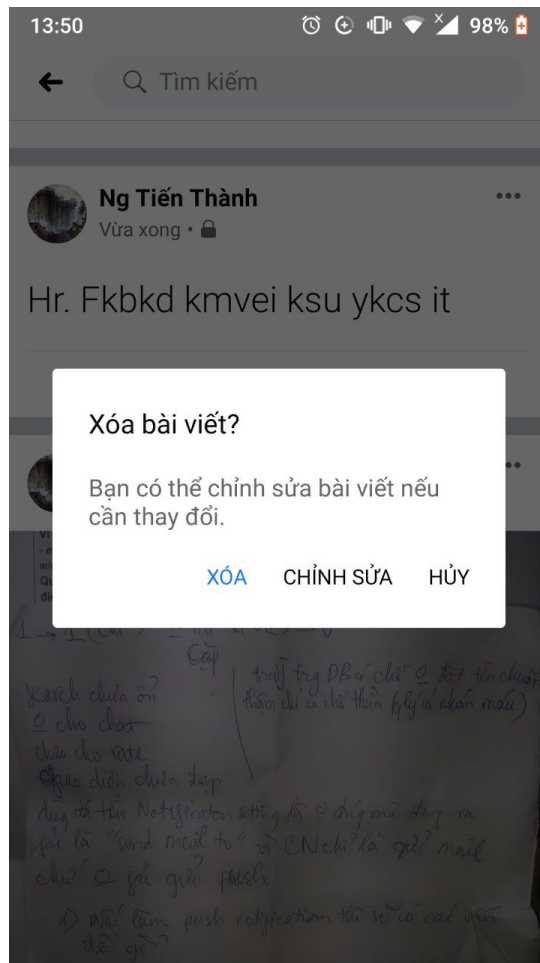
Mục lục

1. Giao diện sửa bài
2. API xóa bài
3. **Giao diện xóa bài**
4. API báo cáo bài
5. Giao diện báo cáo bài
6. API thích bài
7. API xem bình luận bài
8. API đăng bình luận bài
9. Giao diện xem bình luận

Giao diện xóa bài

Giao diện này được xuất hiện sau khi người dùng chọn lấy mục xóa trong danh sách các chức năng nâng cao (ở góc trên tay phải mỗi bài viết)

Khi được chọn, chức năng này yêu cầu người dùng phải khẳng định một lần nữa để đảm bảo chắc chắn



Mục lục

1. Giao diện sửa bài
2. API xóa bài
3. Giao diện xóa bài
4. **API báo cáo bài**
5. Giao diện báo cáo bài
6. API thích bài
7. API xem bình luận bài
8. API đăng bình luận bài
9. Giao diện xem bình luận

report_post

Input

NO	Tên parameter	Type	NN	Mô tả
1	token	string	0	
2	id	integer	0	
3	subject	string	0	
4	details	string	0	

Output

NO	Tên parameter	Type	NN	Mô tả
1	code	string	0	(tham khảo phần response common)
2	message	string	0	

Mô tả report_post

API thực hiện việc báo cáo một bài viết của một tài khoản của người dùng

Request dạng POST

Tham số: **token**, **id** (của bài viết), phân loại nguyên nhân báo cáo và nội dung chi tiết báo cáo

Kết quả đầu ra: Nếu thành công thì mã thông báo thành công được trả về, bài viết được xem xét. Nếu không thành công thì sẽ có các thông báo lỗi tương ứng

Các test case cho report_post

1. Người dùng truyền đúng mã phiên đăng nhập, id bài viết, phân loại vấn đề và chi tiết nội dung

Kết quả mong đợi: 1000 | OK (Thông báo thành công), gửi cho ứng dụng các thông tin cần thiết.

2. Người dùng gửi sai mã phiên đăng nhập (mã bị trống hoặc quá ngắn hoặc mã phiên đăng nhập cũ).

Kết quả mong đợi: ứng dụng sẽ phải đẩy người dùng sang trang đăng nhập. Xem lại test case 3 của change_info_after_signup

Các test case cho report_post (2)

3. Người dùng truyền đúng các thông tin. Nhưng bài viết đã bị khóa (do vi phạm tiêu chuẩn cộng đồng hoặc bị hạn chế tại quốc gia) trước khi gửi báo cáo (trong lúc viết báo cáo vẫn có tồn tại)

Kết quả mong đợi: mã lỗi 1010 và bài viết bị biến mất trong trang hiện tại. Nếu là trang chủ thì ứng dụng sẽ xóa bài viết đó. Nếu là trang cá nhân thì có thể xóa bài viết đó hoặc làm mới lại trang cá nhân (tùy thuộc tình huống).

4. Người dùng truyền đúng các thông tin. Nhưng người dùng đã bị khóa tài khoản (do hệ thống khóa đi mất).

Kết quả mong đợi: ứng dụng sẽ phải đẩy người dùng sang trang đăng nhập. Xem lại test case 3 của change_info_after_signup

Các test case cho report_post (3)

5. Người dùng truyền đúng mã phiên đăng nhập, id bài viết. Nhưng hệ thống không thể tiếp nhận báo cáo (chẳng hạn CSDL không truy cập được)

Kết quả mong đợi: mã lỗi báo hệ thống không thể xóa được do lỗi DB. Nhưng ở phía ứng dụng, cần hiện thông báo khác, chẳng hạn “Không thể kết nối Internet”

6. Người dùng truyền đúng mã phiên đăng nhập, nhưng sai id bài viết (bài viết với id đó không tồn tại).

Kết quả mong đợi: mã lỗi 9992 và báo bài viết không tồn tại.

Các test case cho report_post (4)

7. Người dùng truyền đúng mã phiên đăng nhập, id bài viết. Nhưng đang kết nối thì mạng Internet bị ngắt

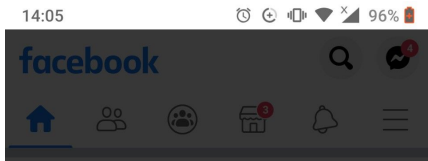
Kết quả mong đợi: ứng dụng cần hiện thông báo “Không thể kết nối Internet” càng sớm càng tốt

CÂU HỎI 2: THEO EM CÒN CÓ CÁC TEST CASE NÀO NỮA?

Mục lục

1. Giao diện sửa bài
2. API xóa bài
3. Giao diện xóa bài
4. API báo cáo bài
5. **Giao diện báo cáo bài**
6. API thích bài
7. API xem bình luận bài
8. API đăng bình luận bài
9. Giao diện xem bình luận

Giao diện báo cáo



Lưu bài viết
Thêm vào danh sách các mục đã lưu.

Ẩn bài viết
Ẩn bớt các bài viết tương tự.

Tạm ẩn Dong trong 30 ngày
Tạm thời dừng xem bài viết.

Ẩn tất cả từ Dong
Không xem bài viết từ người này nữa.

Xem thêm

Tại sao tôi nhìn thấy bài viết này?

Tìm hỗ trợ hoặc báo cáo bài viết
Tôi lo ngại về bài viết này.

Bật thông báo về bài viết này

Sao chép liên kết

Khi hiện ra menu nâng cao ở mỗi bài viết:

Thay “Tìm hỗ trợ hoặc báo cáo bài viết”
thành “Báo cáo bài viết”

Những tùy chọn nào không có API hỗ trợ thì
không cần thiết phải hiển thị

Danh sách các loại báo cáo cần được bắt
chước như facebook.

14:06



Vui lòng chọn vấn đề để tiếp tục

Bạn có thể báo cáo bài viết sau khi chọn vấn đề.

Ảnh khóa thân

Bạo lực

Quấy rối

Tự tử/Tự gây thương tích

Tin giả

Spam

Bán hàng trái phép

Ngôn từ gây thù ghét

Khủng bố

Vấn đề khác

Các bước khác mà bạn có thể thực hiện

Chặn Phạm
Các bạn sẽ không thể nhìn thấy hoặc liên hệ với nhau.

Bỏ theo dõi Phạm
Dừng xem bài viết nhưng vẫn là bạn bè.



Nếu bạn nhận thấy ai đó đang gặp nguy hiểm, đừng chần chừ mà hãy báo ngay cho dịch vụ cấp cứu tại địa phương.

Tiếp

Giao diện báo cáo (2)



14:06 96%

Vui lòng chọn vấn đề để tiếp tục

Bạn có thể báo cáo bài viết sau khi chọn vấn đề.

- Ảnh khóa thân
- Bạo lực
- Quấy rối
- Tự tử/Tự gây thương tích
- Tin giả
- Spam
- Bán hàng trái phép
- Ngôn từ gây thù ghét
- Khủng bố
- Vấn đề khác

Các bước khác mà bạn có thể thực hiện

-  **Chặn Phạm**
Các bạn sẽ không thể nhìn thấy hoặc liên hệ với nhau.
-  **Bỏ theo dõi Phạm**
Đừng xem bài viết nhưng vẫn là bạn bè.



Nếu bạn nhận thấy ai đó đang gặp nguy hiểm, đừng chần chừ mà hãy báo ngay cho dịch vụ cấp cứu tại địa phương.

Tiếp

Mục ở trên cùng sẽ là phân loại báo cáo, còn mục ở dưới sẽ là nội dung chi tiết

Có các tùy chọn “Chỉ chặn trên Messenger” và “Bỏ theo dõi” sẽ không cần cài đặt

Chặn một người sẽ chặn ở trên cả Facebook và tin nhắn chat.

14:06

95%



Bạn đã chọn

- ✓ Ngôn từ gây thù ghét
- ✓ Thành phần tôn giáo

Bạn có thể báo cáo nếu cho rằng nội dung này vi phạm **Tiêu chuẩn cộng đồng** của chúng tôi. Xin lưu ý rằng đội ngũ xét duyệt của chúng tôi hiện có ít nhân lực hơn.

Các bước khác mà bạn có thể thực hiện

-  **Chỉ chặn Phạm trên Messenger**
Các bạn sẽ không nhận được tin nhắn hoặc cuộc gọi của nhau, nhưng vẫn có thể tương tác với nhau trên Facebook.
-  **Chặn Phạm**
Các bạn sẽ không thể nhìn thấy hoặc liên hệ với nhau.
-  **Bỏ theo dõi Phạm**
Đừng xem bài viết nhưng vẫn là bạn bè.

Xong

Giao diện báo cáo (3)


14:06 96%


Vui lòng chọn vấn đề để tiếp tục


Bạn có thể báo cáo bài viết sau khi chọn vấn đề.

- Ảnh khóa thân
- Bạo lực
- Quấy rối
- Tự tử/Tự gây thương tích
- Tin giả
- Spam
- Bán hàng trái phép
- Ngôn từ gây thù ghét
- Khủng bố
- Vấn đề khác

Các bước khác mà bạn có thể thực hiện

 **Chặn Phạm**
Các bạn sẽ không thể nhìn thấy hoặc liên hệ với nhau.

 **Bỏ theo dõi Phạm**
Đừng xem bài viết nhưng vẫn là bạn bè.

 Nếu bạn nhận thấy ai đó đang gặp nguy hiểm, đừng chần chừ mà hãy báo ngay cho dịch vụ cấp cứu tại địa phương.

Tiếp

Nhóm cần hoàn thiện việc mô phỏng đúng tất cả các loại vấn đề trong báo cáo như ở bản nguyên thủy.

14:10	
← Tìm kiếm	
Ví dụ	
Quyền sở hữu trí tuệ	>
Gian lận hoặc lừa đảo	>
Chế giễu nạn nhân	>
Bắt nạt	>
Lạm dụng trẻ em	>
Ngược đãi động vật	>
Hoạt động tình dục	>
Tự tử/Tự gây thương tích	>
Ngôn từ gây thù ghét	>
Quảng bá hành vi sử dụng ma túy	>
Hình ảnh thân mật không có sự đồng thuận	>

Mục lục

1. Giao diện sửa bài
2. API xóa bài
3. Giao diện xóa bài
4. API báo cáo bài
5. Giao diện báo cáo bài
6. **API thích bài**
7. API xem bình luận bài
8. API đăng bình luận bài
9. Giao diện xem bình luận

like

Input				
NO	Tên parameter	Type	NN	Mô tả
1	token	string	0	
2	id	integer	0	id
Output				
NO	Tên parameter	Type	NN	Mô tả
1	code	string	0	(tham khảo phần response common)
2	message	string	0	
3	data	string	0	
	like	string	0	số like hiện tại

Mô tả like

API thực hiện thích hoặc bỏ thích một bài viết của một người dùng

Request dạng POST

Tham số: **token, id** (của bài viết)

Kết quả đầu ra: Nếu thành công thì mã thông báo thành công được trả về, bài viết được tăng/giảm thêm một like của chính người dùng. Nếu không thành công thì sẽ có các thông báo lỗi tương ứng

Các test case cho like

1. Người dùng truyền đúng mã phiên đăng nhập, id bài viết

Kết quả mong đợi: 1000 | OK (Thông báo thành công), cơ sở dữ liệu trên phía server sẽ có các cập nhật về danh sách like, còn phía ứng dụng sẽ có các cập nhật giao diện cần thiết (đổi màu và đổi danh sách like).

2. Người dùng gửi sai mã phiên đăng nhập (mã bị trống hoặc quá ngắn hoặc mã phiên đăng nhập cũ).

Kết quả mong đợi: ứng dụng sẽ phải đẩy người dùng sang trang đăng nhập. Xem lại test case 3 của `change_info_after_signup`

Các test case cho like (2)

3. Người dùng truyền đúng các thông tin. Nhưng bài viết đã bị khóa (do vi phạm tiêu chuẩn cộng đồng hoặc bị hạn chế tại quốc gia) trước khi gửi báo cáo (trong lúc viết báo cáo vẫn có tồn tại)

Kết quả mong đợi: mã lỗi 1010 và bài viết bị biến mất trong trang hiện tại. Nếu là trang chủ thì ứng dụng sẽ xóa bài viết đó. Nếu là trang cá nhân thì có thể xóa bài viết đó hoặc làm mới lại trang cá nhân (tùy thuộc tình huống).

4. Người dùng truyền đúng các thông tin. Nhưng người dùng đã bị khóa tài khoản (do hệ thống khóa đi mất).

Kết quả mong đợi: ứng dụng sẽ phải đẩy người dùng sang trang đăng nhập. Xem lại test case 3 của `change_info_after_signup`

Các test case cho like (3)

5. Người dùng truyền đúng mã phiên đăng nhập, id bài viết. Nhưng hệ thống không thể tiếp nhận báo cáo (chẳng hạn CSDL không truy cập được)

Kết quả mong đợi: mã lỗi báo hệ thống không thể xóa được do lỗi DB. Nhưng ở phía ứng dụng, cần hiện thông báo khác, chẳng hạn “Không thể kết nối Internet”

6. Người dùng truyền đúng mã phiên đăng nhập, nhưng sai id bài viết (bài viết với id đó không tồn tại).

Kết quả mong đợi: mã lỗi 9992 và báo bài viết không tồn tại.

Các test case cho like (4)

7. Người dùng truyền đúng mã phiên đăng nhập, id bài viết. Nhưng đang kết nối thì mạng Internet bị ngắt

Kết quả mong đợi: ứng dụng cần hiện thông báo “Không thể kết nối Internet” càng sớm càng tốt

8. Người dùng truyền đúng mã phiên đăng nhập, id bài viết. Nhưng server trả về với số lượng like bị lỗi (giá trị âm hoặc quá lớn hàng tỉ)

Kết quả mong đợi: ứng dụng cần thay đổi giao diện cho phù hợp với hành động của người dùng lên trang. Số lượng like có thể hiển thị rằng “Bạn thích bài viết” (nếu hành động like) hoặc không có ai like (nếu hành động là bỏ like)

Các test case cho like (5)

9. Người dùng truyền đúng mã phiên đăng nhập, id bài viết. Nhưng server trả về với số lượng like bị sai logic (chẳng hạn người dùng thực hiện thích nhưng số like trả về bằng 0)

Kết quả mong đợi: ứng dụng cần thay đổi giao diện cho phù hợp với hành động của người dùng lên trang. Số lượng like có thể vẫn hiển thị rằng “Bạn thích bài viết” (nếu hành động like)

CÂU HỎI 3: THEO EM CÒN CÓ CÁC TEST CASE NÀO NỮA?

Mục lục

1. Giao diện sửa bài
2. API xóa bài
3. Giao diện xóa bài
4. API báo cáo bài
5. Giao diện báo cáo bài
6. API thích bài
7. **API xem bình luận bài**
8. API đăng bình luận bài
9. Giao diện xem bình luận

get_comment

Input

NO	Tên parameter	Type	NN	Mô tả
1	id	string	0	id của bài viết
2	index	string	0	
3	count	string	0	

Output

NO	Tên parameter	Type	NN	Mô tả
1	code	string	0	(tham khảo phản response common)
2	message	string	0	
3	data	array	0	
	id	string	0	
	comment	string	0	
	created	string	0	
	poster	array	0	thông tin người viết comment
	id	string	0	
	name	string	0	
	avatar	string	0	
4	is_blocked	array	0	

Mô tả get_comment

API thực hiện lấy danh sách comments của một bài viết của một người dùng

Request dạng POST

Tham số: **token**, **id** (của bài viết), index, count (để lấy danh sách theo từng phần)

Kết quả đầu ra: Nếu thành công thì mã thông báo thành công được trả về, popup bình luận hiện ra danh sách các bình luận mới nhất. Nếu có nhiều bình luận thì sẽ có tùy chọn để tải thêm các bình luận cũ hơn

Các test case cho get_comment

1. Người dùng truyền đúng mã phiên đăng nhập, id bài viết, các chỉ số khác đúng

Kết quả mong đợi: 1000 | OK (Thông báo thành công), hiển thị ra danh sách các bình luận.

2. Người dùng gửi sai mã phiên đăng nhập (mã bị trống hoặc quá ngắn hoặc mã phiên đăng nhập cũ).

Kết quả mong đợi: ứng dụng sẽ phải đẩy người dùng sang trang đăng nhập. Xem lại test case 3 của change_info_after_signup

Các test case cho get_comment(2)

3. Người dùng truyền đúng các thông tin. Nhưng bài viết đã bị khóa (do vi phạm tiêu chuẩn cộng đồng hoặc bị hạn chế tại quốc gia) trước khi gửi yêu cầu (trong lúc gửi yêu cầu xem bình luận thì bài viết vẫn có tồn tại)

Kết quả mong đợi: mã lỗi 1010 và bài viết bị biến mất trong trang hiện tại. Nếu là trang chủ thì ứng dụng sẽ xóa bài viết đó. Nếu là trang cá nhân thì có thể xóa bài viết đó hoặc làm mới lại trang cá nhân (tùy thuộc tình huống).

4. Người dùng truyền đúng các thông tin. Nhưng người dùng đã bị khóa tài khoản (do hệ thống khóa đi mất).

Kết quả mong đợi: ứng dụng sẽ phải đẩy người dùng sang trang đăng nhập. Xem lại test case 3 của change_info_after_signup

Các test case cho get_comment (3)

5. Người dùng truyền đúng mã phiên đăng nhập, id bài viết cũng như các tham số khác. Nhưng hệ thống không thể lấy thêm danh sách các bình luận (chẳng hạn CSDL không truy cập được)

Kết quả mong đợi: mã lỗi báo hệ thống không thể xóa được do lỗi DB. Nhưng ở phía ứng dụng, cần hiện thông báo khác, chẳng hạn “Không thể kết nối Internet”

6. Người dùng truyền đúng mã phiên đăng nhập, nhưng sai id bài viết (bài viết với id đó không tồn tại).

Kết quả mong đợi: mã lỗi 9992 và báo bài viết không tồn tại.

Các test case cho get_comment (4)

7. Người dùng truyền đúng mã phiên đăng nhập, id bài viết. Nhưng đang kết nối thì mạng Internet bị ngắt

Kết quả mong đợi: ứng dụng cần hiện thông báo “Không thể kết nối Internet” càng sớm càng tốt

8. Người dùng truyền đúng các thông số. Nhưng hệ thống chỉ còn số bình luận ít hơn số count.

Kết quả mong đợi: ứng dụng cần hiển thị các bình luận còn lại, nhưng chắc chắn không còn bình luận nào thêm nữa, hệ thống sẽ không có câu “Tải thêm các bình luận...”

Các test case cho get_comment (5)

9. Người dùng truyền đúng mã phiên đăng nhập, id bài viết cũng như các tham số khác. Nhưng trong danh sách các bình luận có những người đang chặn người dùng hoặc bị chính người dùng chặn

Kết quả mong đợi: server cần lọc ra và đếm danh sách các bình luận còn lại.
Nếu không có bình luận nào thì cần gửi mã lỗi 9994

10. Người dùng truyền đúng mã phiên đăng nhập, đúng id bài viết, nhưng hai tham số còn lại bị sai

Kết quả mong đợi: mã lỗi báo sai tham số, hệ thống giữ nguyên việc hiển thị mà không báo cho người dùng biết

Các test case cho get_comment (6)

11. Người dùng truyền đúng mã phiên đăng nhập, id bài viết cũng như các tham số khác. Nhưng trong quá trình xem bình luận thì hệ thống nhận yêu cầu (của chủ bài viết) chặn người dùng

Kết quả mong đợi: server báo về cho client. Ứng dụng phát hiện ra việc chặn, thực hiện ẩn popup bình luận, xóa bài viết khỏi danh sách

12. Người dùng truyền đúng mã phiên đăng nhập, id bài viết cũng như các tham số khác. Nhưng trong quá trình xem bình luận thì hệ thống nhận yêu cầu (của một trong số những người tham gia bình luận) chặn người dùng

Kết quả mong đợi: từ đây trở đi, server không gửi về các bình luận của người chủ động chặn. Bình luận khác (đã hiện ở client) vẫn tạm cho hiển thị nốt lần đó.

Các test case cho get_comment (7)

CÂU HỎI 4: THEO EM CÒN CÓ CÁC TEST CASE NÀO NỮA?

Mục lục

1. Giao diện sửa bài
2. API xóa bài
3. Giao diện xóa bài
4. API báo cáo bài
5. Giao diện báo cáo bài
6. API thích bài
7. API xem bình luận bài
8. **API đăng bình luận bài**
9. Giao diện xem bình luận

set_comment

Input

NO	Tên parameter	Type	NN	Mô tả
1	token	string	0	
2	id	string	0	id của bài viết
3	comment	string	0	
4	index	string	0	chỉ số comment bắt đầu (phục vụ lấy comment về)
5	count	string	0	số lượng comment (phục vụ lấy comment về)

Output

NO	Tên parameter	Type	NN	Mô tả
1	code	string	0	(tham khảo phần response common)
2	message	string	0	
3	data	array	0	
	id	string	X	thông tin người viết comment
	comment	string	X	
	created	string	X	
	poster	array	X	
	id	string	X	
	name	string	X	
	avatar	string	X	
4	is_blocked	string	X	

Mô tả set_comment

API thực hiện đăng thêm một comment của vào một bài viết

Request dạng POST

Tham số: **token**, **id** (của bài viết), nội dung comment, index, count (để lấy danh sách theo từng phần)

Kết quả đầu ra: Nếu thành công thì mã thông báo thành công được trả về, popup bình luận hiện ra danh sách các bình luận mới nhất. Nếu có nhiều bình luận thì sẽ có tùy chọn để tải thêm các bình luận cũ hơn, nếu có thêm các bình luận mới thì sẽ hiển thị chúng ra. Chú ý rằng giao diện phải cuộn xuống bình luận mới nhất

Các test case cho set_comment

1. Người dùng truyền đúng mã phiên đăng nhập, id bài viết, các chỉ số khác đúng. Hệ thống không có bình luận mới nào mà ứng dụng chưa lấy về.

Kết quả mong đợi: 1000 | OK (Thông báo thành công), hiển thị ra bình luận mới nhất của người dùng.

2. Người dùng gửi sai mã phiên đăng nhập (mã bị trống hoặc quá ngắn hoặc mã phiên đăng nhập cũ), các thông số khác phù hợp.

Kết quả mong đợi: ứng dụng sẽ phải đẩy người dùng sang trang đăng nhập. Xem lại test case 3 của change_info_after_signup

Các test case cho set_comment (2)

3. Người dùng truyền đúng các thông tin. Nhưng bài viết đã bị khóa (do vi phạm tiêu chuẩn cộng đồng hoặc bị hạn chế tại quốc gia) trước khi gửi bình luận mới (trong lúc viết bình luận thì bài viết vẫn có tồn tại)

Kết quả mong đợi: mã lỗi 1010 và bài viết bị biến mất trong trang hiện tại. Nếu là trang chủ thì ứng dụng sẽ xóa bài viết đó. Nếu là trang cá nhân thì có thể xóa bài viết đó hoặc làm mới lại trang cá nhân (tùy thuộc tình huống).

4. Người dùng truyền đúng các thông tin. Nhưng người dùng đã bị khóa tài khoản (do hệ thống khóa đi mất).

Kết quả mong đợi: ứng dụng sẽ phải đẩy người dùng sang trang đăng nhập. Xem lại test case 3 của change_info_after_signup

Các test case cho set_comment (3)

5. Người dùng truyền đúng mã phiên đăng nhập, id bài viết cũng như các tham số khác. Nhưng hệ thống không thể chèn thêm vào danh sách các bình luận (chẳng hạn CSDL không truy cập được)

Kết quả mong đợi: mã lỗi báo hệ thống không thể xóa được do lỗi DB. Nhưng ở phía ứng dụng, cần hiện thông báo khác, chẳng hạn “Không thể kết nối Internet”

6. Người dùng truyền đúng mã phiên đăng nhập, nhưng sai id bài viết (bài viết với id đó không tồn tại).

Kết quả mong đợi: mã lỗi 9992 và báo bài viết không tồn tại.

Các test case cho set_comment (4)

7. Người dùng truyền đúng mã phiên đăng nhập, id bài viết. Nhưng đang kết nối thì mạng Internet bị ngắt

Kết quả mong đợi: ứng dụng cần hiện thông báo “Không thể kết nối Internet” càng sớm càng tốt

8. Người dùng truyền đúng các thông số. Và hệ thống có trả về thêm một số bình luận mới nữa.

Kết quả mong đợi: ứng dụng cần hiển thị các bình luận còn lại, và các bình luận này sẽ xuất hiện ở bên dưới, ứng dụng cần phải tự động cuộn xuống dưới bình luận cuối cùng.

Các test case cho set_comment (5)

9. Người dùng truyền đúng mã phiên đăng nhập, id bài viết cũng như các tham số khác. Nhưng trong danh sách các bình luận mới (được API này trả về) có những người đang chặn người dùng hoặc bị chính người dùng chặn

Kết quả mong đợi: server cần lọc ra và đếm danh sách các bình luận còn lại. Nếu không có bình luận nào thì vẫn báo là bình luận thành công.

10. Người dùng truyền đúng mã phiên đăng nhập, đúng id bài viết, và có nội dung bình luận nhưng hai tham số còn lại bị sai (hoặc nội dung bình luận trống)

Kết quả mong đợi: ứng dụng cần phải tự kiểm tra trước khi gửi, nhưng nếu vì lý do nào đó vẫn gửi lên được thì server cần trả về mã lỗi báo sai tham số, hệ thống giữ nguyên việc hiển thị mà không báo cho người dùng biết

Các test case cho set_comment (6)

11. Người dùng truyền đúng mã phiên đăng nhập, id bài viết cũng như các tham số khác. Nhưng trong quá trình đăng bình luận thì hệ thống nhận yêu cầu (của chủ bài viết) chặn người dùng

Kết quả mong đợi: server báo về cho client. Ứng dụng phát hiện ra việc chặn, thực hiện ẩn popup bình luận, xóa bài viết khỏi danh sách

12. Người dùng truyền đúng mã phiên đăng nhập, id bài viết cũng như các tham số khác. Nhưng trong quá trình đăng bình luận thì hệ thống nhận yêu cầu (của một trong số những người tham gia bình luận) chặn người dùng

Kết quả mong đợi: từ đây trở đi, server không gửi về các bình luận của người chủ động chặn. Bình luận khác (đã hiện ở client) vẫn tạm cho hiển thị nốt lần đó.

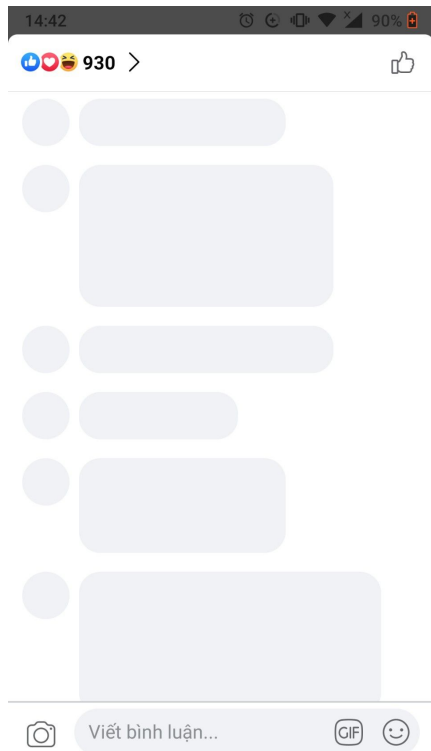
Các test case cho set_comment (7)

CÂU HỎI 5: THEO EM CÒN CÓ CÁC TEST CASE NÀO NỮA?

Mục lục

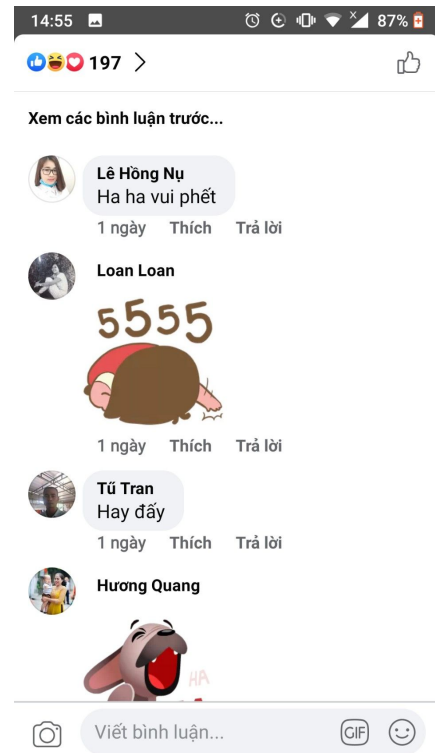
1. Giao diện sửa bài
2. API xóa bài
3. Giao diện xóa bài
4. API báo cáo bài
5. Giao diện báo cáo bài
6. API thích bài
7. API xem bình luận bài
8. API đăng bình luận bài
9. **Giao diện xem bình luận**

Giao diện bình luận

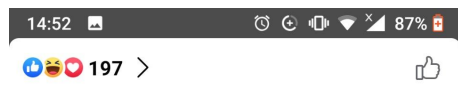


Giao diện này hiển thị các bình luận, khi server chưa trả về kết quả được thì hệ thống hiển thị các **animation** như **Facebook** (các bình luận màu xám thoát ẩn thoát hiện).

Khi đã có các bình luận hiển thị lên, nếu server trả về đúng số lượng yêu cầu trong biến count (mặc định 20) thì ứng dụng vẫn hiển thị lên “Xem các bình luận trước...”



Giao diện bình luận (2)

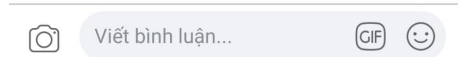


Nếu đang request lấy bình luận ĐẦU TIÊN (tức $\text{index} = 0$) mà mạng kết nối bị ngắt, ứng dụng sẽ hiển thị như hình bên.

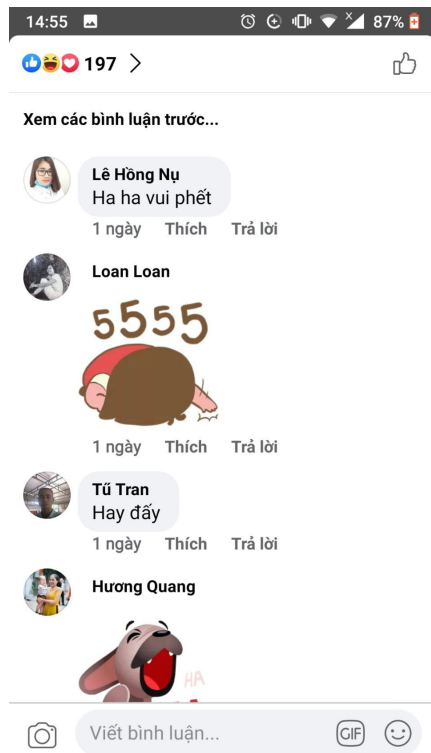
Ứng dụng phải lưu lại các bình luận này khi nào cần sẽ hiển thị lên với vị trí câu bình luận là ở trong ô nhập chữ góc dưới cùng của màn hình.

Các bình luận lưu này chỉ tạm thời ở một thiết bị, sẽ bị xóa nếu người dùng đăng xuất.

Nếu đang request lấy CÁC bình luận tiếp theo (tức $\text{index} \neq 0$) mà mạng kết nối bị ngắt, ứng dụng sẽ vẫn giữ các bình luận cũ và cảnh báo về mạng.



Giao diện bình luận (3)



Việc nhấn vào “Xem các bình luận trước...” chỉ giúp hiển thị các bình luận cũ hơn. Các bình luận mới nhất cần phải hiển thị theo cơ chế gửi API **set_comment** hoặc **push notification**.

Cần chú ý việc hiển thị thời gian của các bình luận.

Nhắc lại, không làm chức năng trả lời bình luận hoặc bày tỏ cảm xúc của bình luận.

Các bình luận chỉ là **text** và chứa được **emoticons**, các biểu tượng khác hoặc hình ảnh/video chưa phải làm

HẾT TUẦN 4