

KẾ HOẠCH LÀM VIỆC TUẦN 3

**Viện Công nghệ thông tin và Truyền thông
2021**

Nhắc lại

Tổng cộng có các API sau:

signup, login, logout, get_user_info, set_user_info, add post,
get post, get list posts, check new item, edit post,
delete post, report_post, set_comment, get_comment, like,
edit_comment, del_comment, search, set_request_friend,
get_requested_friend, set_accept_friend, get_user_friends,
change_password, set_block_user, set_block_diary,
get_conversation, delete_message, get_list_conversation,
delete_conversation

Các API thêm vào:

set_official_account, check_verify_code, del_saved_search,
get_list_suggested_friends, get_verify_code,
get_saved_search,

admin_api: get_admin_permission, get_user_list, set_role,
get_analyst_result, set_user_state, delete_user,
get_user_basic_info

Mục lục

1. API báo cáo bài viết
2. Giao diện báo cáo bài viết
3. API lấy danh sách bình luận
4. Giao diện danh sách bình luận bài viết
5. API bình luận bài viết
6. Giao diện bình luận bài viết
7. API xóa bài viết
8. Giao diện xóa bài viết
9. API chỉnh sửa bình luận
10. Giao diện chỉnh sửa bình luận
11. API thích bài viết

Mục lục

1. **API báo cáo bài viết**
2. Giao diện báo cáo bài viết
3. API lấy danh sách bình luận
4. Giao diện danh sách bình luận bài viết
5. API bình luận bài viết
6. Giao diện bình luận bài viết
7. API xóa bài viết
8. Giao diện xóa bài viết
9. API chỉnh sửa bình luận
10. Giao diện chỉnh sửa bình luận
11. API thích bài viết

Report_post

Input

NO	Tên parameter	Type	NN	Mô tả
1	token	string	0	
2	id	string	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), **subject** (là con số bắt đầu từ 0) nguyên nhân báo cáo và **details** chi tiết báo cáo.

Kết quả đầu ra: 1000|OK. 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)

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.

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: 1010 | 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).

Các test case cho report_post (3)

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.
5. 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.

Các test case cho report_post (4)

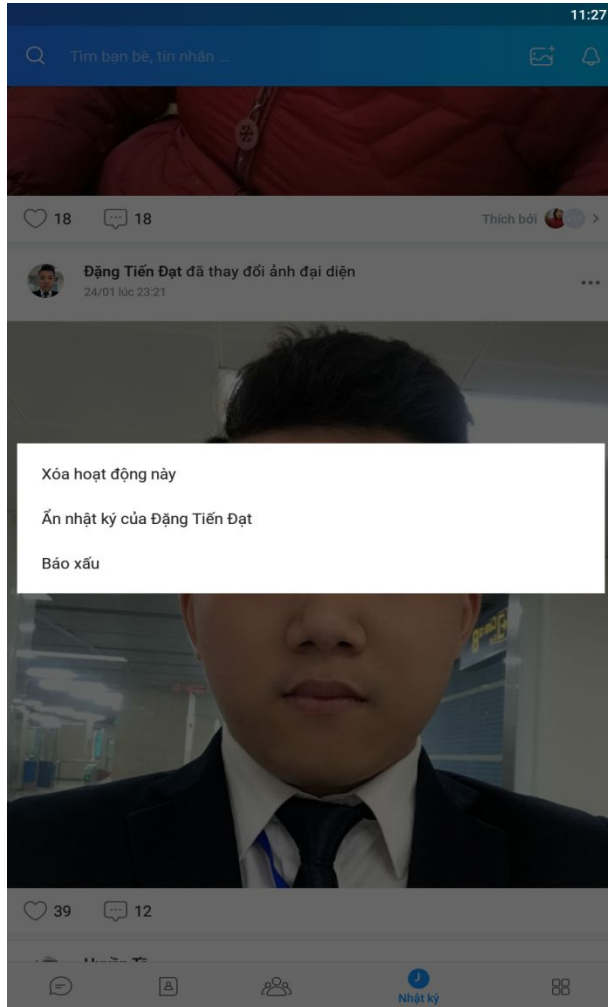
6. Người dùng truyền đúng mã phiên đăng nhập, id bài viết. Nhưng hệ thống không thể tiếp nhận báo cáo (chẳng hạn CSDL không truy cập được).
Kết quả mong đợi: mã lỗi báo hệ thống không thể xóa được do lỗi DB. Nhưng ở phía ứng dụng, cần hiện thông báo khác, chẳng hạn “**Không thể kết nối Internet**”
7. 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.

Mục lục

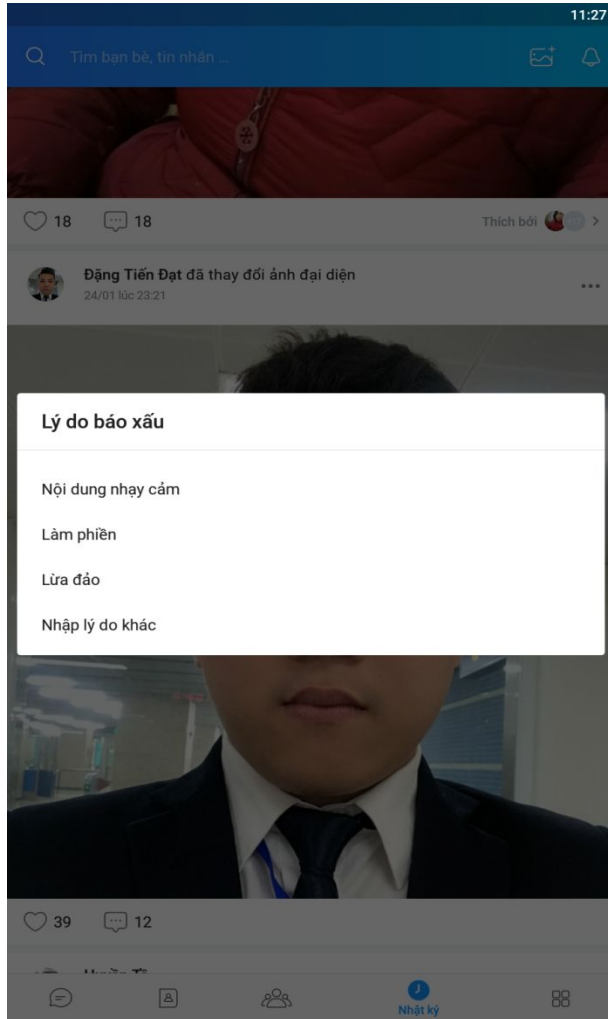
1. API báo cáo bài viết
2. Giao diện báo cáo bài viết
3. API lấy danh sách bình luận
4. Giao diện danh sách bình luận bài viết
5. API bình luận bài viết
6. Giao diện bình luận bài viết
7. API xóa bài viết
8. Giao diện xóa bài viết
9. API chỉnh sửa bình luận
10. Giao diện chỉnh sửa bình luận
11. API thích bài viết

Giao diện báo cáo bài viết (1)

Khi hiện ra menu nâng cao ở mỗi bài viết, người dùng chọn **Báo xấu** để báo cáo bài viết.



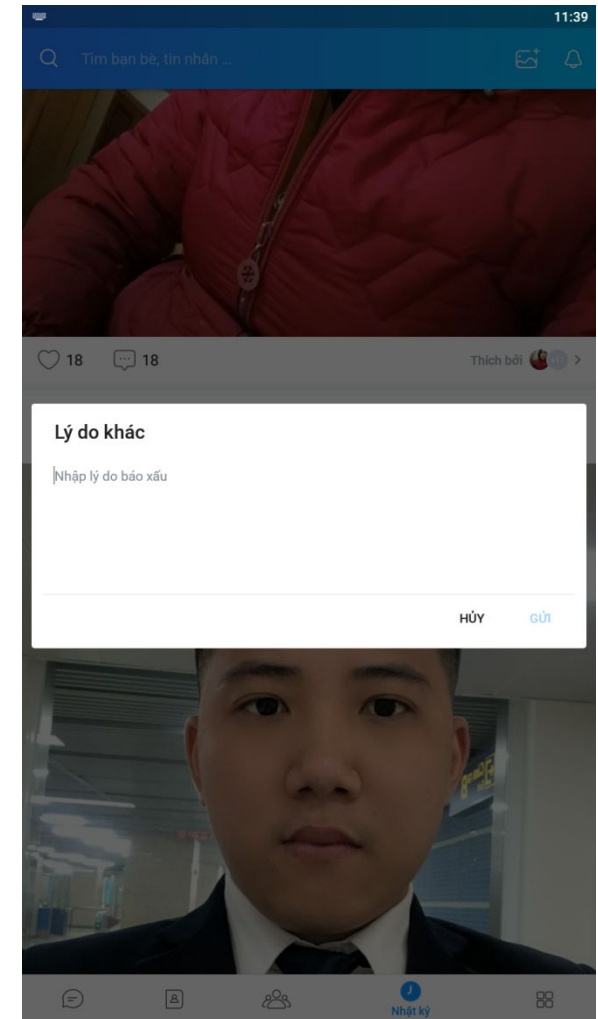
Giao diện báo cáo bài viết (2)



Có 4 lựa chọn báo cáo là: nội dung nhạy cảm, làm phiền, lừa đảo, lý do khác.

Nếu chọn **lý do khác** thì sẽ hiện lên form nhập thông tin để gửi đi.

Báo cáo thành công sẽ hiển thị thông báo “Báo cáo thành công”



Mục lục

1. API báo cáo bài viết
2. Giao diện báo cáo bài viết
3. **API lấy danh sách bình luận**
4. Giao diện danh sách bình luận bài viết
5. API bình luận bài viết
6. Giao diện bình luận bài viết
7. API xóa bài viết
8. Giao diện xóa bài viết
9. API chỉnh sửa bình luận
10. Giao diện chỉnh sửa bình luận
11. API thích bài viết

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	
4	token	string	0	

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	
	id	string	0	
	comment	string	0	
	created	string	0	
	poster	string	0	thông tin người viết comment
	id	string	0	
	name	string	0	
	avatar	string	0	
4	is_blocked	string	0	

Mô tả get_comment(1)

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), trong trang trước không nói rõ nhưng API này có thêm tham số **token**

Mô tả `get_comment(2)`

Kết quả đầu ra: 1000 | OK - 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.

Trường **`is_blocked`** trả về 1 nếu người dùng (người đang xem bình luận) bị người chủ bài viết chặn lại.

Các test case cho get_comment (1)

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.

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).

Các test case cho get_comment (3)

4. 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”

Các test case cho get_comment (4)

5. 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.
6. 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.

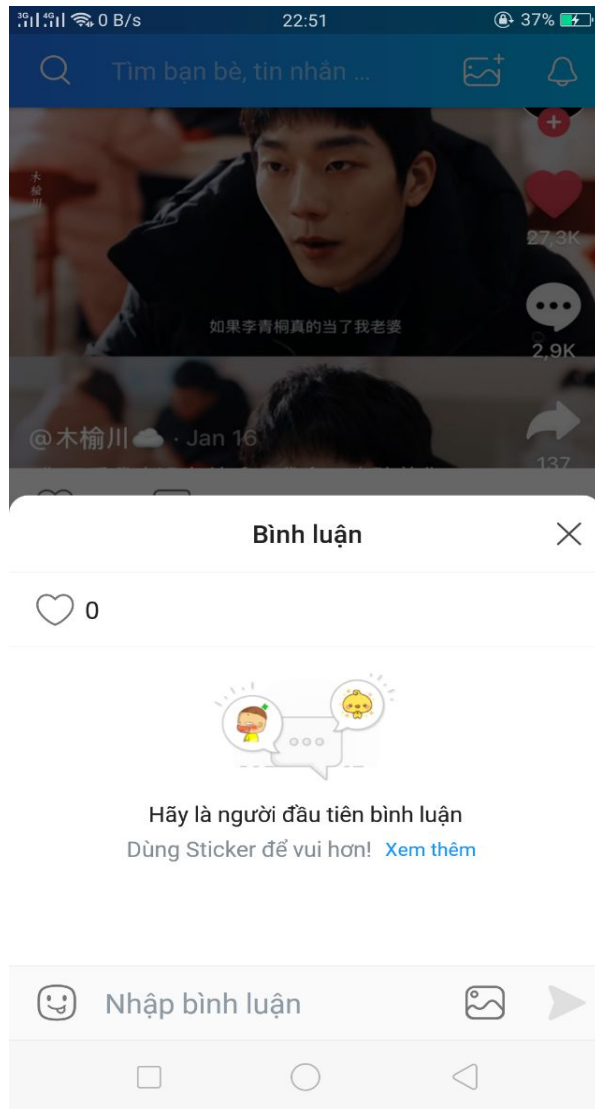
Các test case cho get_comment (5)

7. 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...”
8. 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.

Mục lục

1. API báo cáo bài viết
2. Giao diện báo cáo bài viết
3. API lấy danh sách bình luận
4. **Giao diện danh sách bình luận bài viết**
5. API bình luận bài viết
6. Giao diện bình luận bài viết
7. API xóa bài viết
8. Giao diện xóa bài viết
9. API chỉnh sửa bình luận
10. Giao diện chỉnh sửa bình luận
11. API thích bài viết

Giao diện danh sách bình luận bài viết



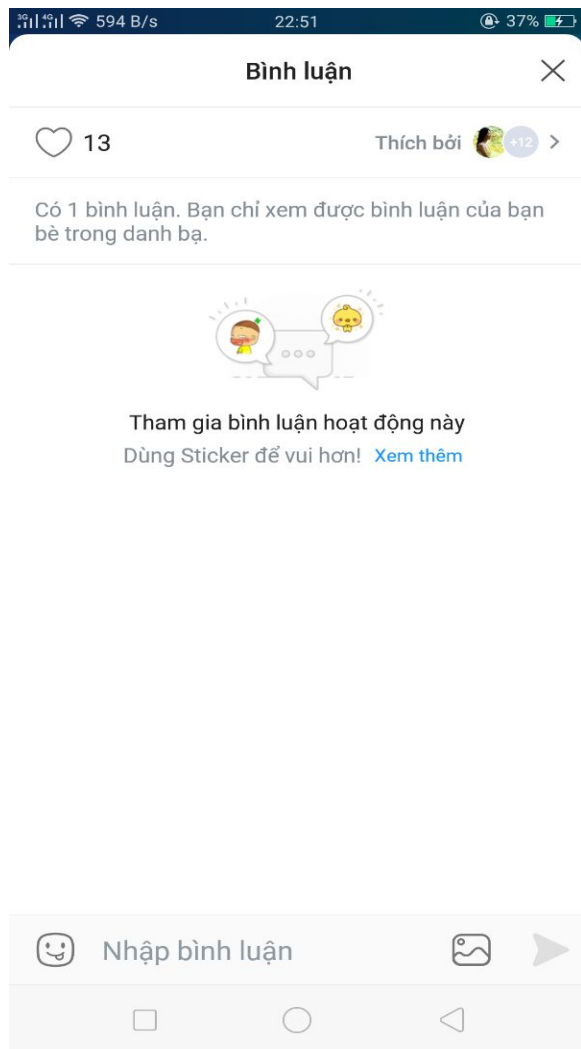
Giao diện này hiển thị các bình luận.

Khi server trả về các bình luận thì khi số lượng trả về vượt quá **count** (mặc định 20) thì sẽ thêm một lựa chọn là “Xem các bình luận trước”.

Gửi kết quả trả về trong API [set_comment](#) hoặc [push_notification](#).

Các bình luận chỉ là [text](#) và có thể chứa các [emotions](#).

Giao diện danh sách bình luận bài viết



Khi người dùng khác không trong danh sách bạn bè sẽ không thể xem được danh sách bình luận bài viết và cũng không thể bình luận.

Mục lục

1. API báo cáo bài viết
2. Giao diện báo cáo bài viết
3. API lấy danh sách bình luận
4. Giao diện danh sách bình luận bài viết
5. **API bình luận bài viết**
6. Giao diện bình luận bài viết
7. API xóa bài viết
8. Giao diện xóa bài viết
9. API chỉnh sửa bình luận
10. Giao diện chỉnh sửa bình luận
11. API thích bài viết

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	string	0	
	id	string	X	
	comment	string	X	
	created	string	X	
	poster	string	X	thông tin người viết comment
	id	string	X	
	name	string	X	
	avatar	string	X	
4	is_blocked	string	X	

Mô tả set_comment (1)

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

Request dạng: **POST**

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

Mô tả `set_comment` (2)

Kết quả đầu ra: 1000|OK - 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

Trường **`is_blocked`** trả về 1 nếu người dùng (người đang xem bình luận) bị người chủ bài viết chặn lại

Các test case cho set_comment (1)

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.

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).

Các test case cho set_comment (3)

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.
5. 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)

6. 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”

Các test case cho set_comment (5)

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.

Mục lục

1. API báo cáo bài viết
2. Giao diện báo cáo bài viết
3. API lấy danh sách bình luận
4. Giao diện danh sách bình luận bài viết
5. API bình luận bài viết
6. **Giao diện bình luận bài viết**
7. API xóa bài viết
8. Giao diện xóa bài viết
9. API chỉnh sửa bình luận
10. Giao diện chỉnh sửa bình luận
11. API thích bài viết

Giao diện bình luận bài viết

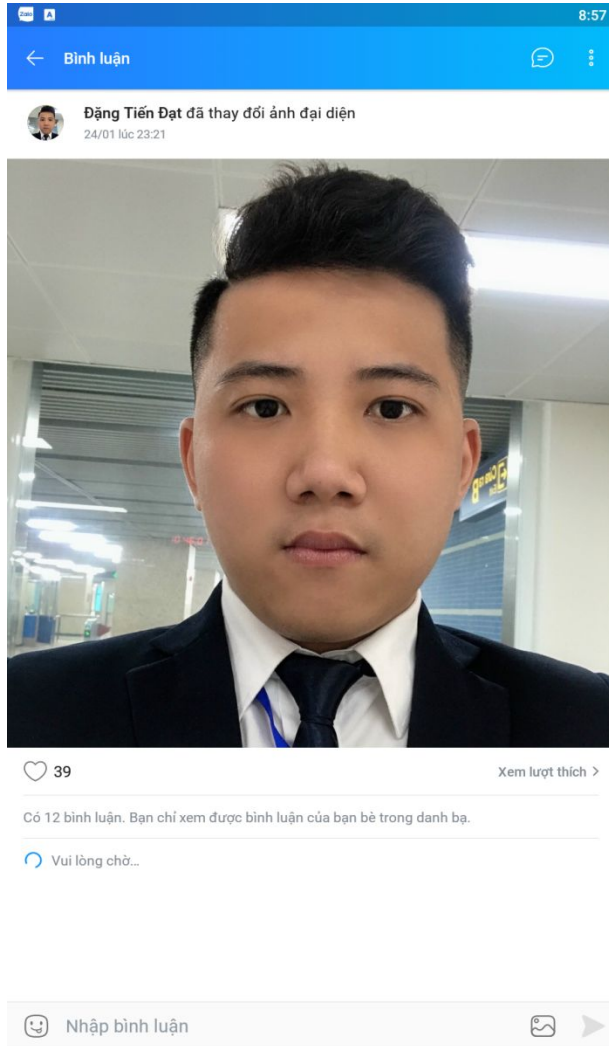


Phần bình luận chỉ hiển thị bình luận của các bạn bè trong danh sách bạn bè

Nếu chưa có bình luận nào hoặc có nhưng bình luận không phải của bạn bè người dùng thì hiển thị như hình bên phải



Giao diện bình luận bài viết



Nếu đang request bình luận mà bị ngắt kết nối mạng thì sẽ hiển thị dòng chữ “Vui lòng chờ...” như hình bên

Mục lục

1. API báo cáo bài viết
2. Giao diện báo cáo bài viết
3. API lấy danh sách bình luận
4. Giao diện danh sách bình luận bài viết
5. API bình luận bài viết
6. Giao diện bình luận bài viết
7. **API xóa bình luận**
8. Giao diện xóa bài viết
9. API chỉnh sửa bình luận
10. Giao diện chỉnh sửa bình luận
11. API thích bài viết

del_comment

Input

NO	Tên parameter	Type	NN	Mô tả
1	token	string	0	
2	id	string	0	id của bài viết
3	id_com	string	0	id của bình luận

Output

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

Mô tả del_comment

API thực hiện việc xóa một bình luận của bài viết

Request dạng: **POST**

Tham số: **token**, **id** (của bài viết), **id_com** (vị trí của comment trong bài viết).

Kết quả đầu ra: 1000|OK - 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 del_comment (1)

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.

Các test case cho del_comment (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).

Các test case cho del_comment (3)

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.
5. 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.

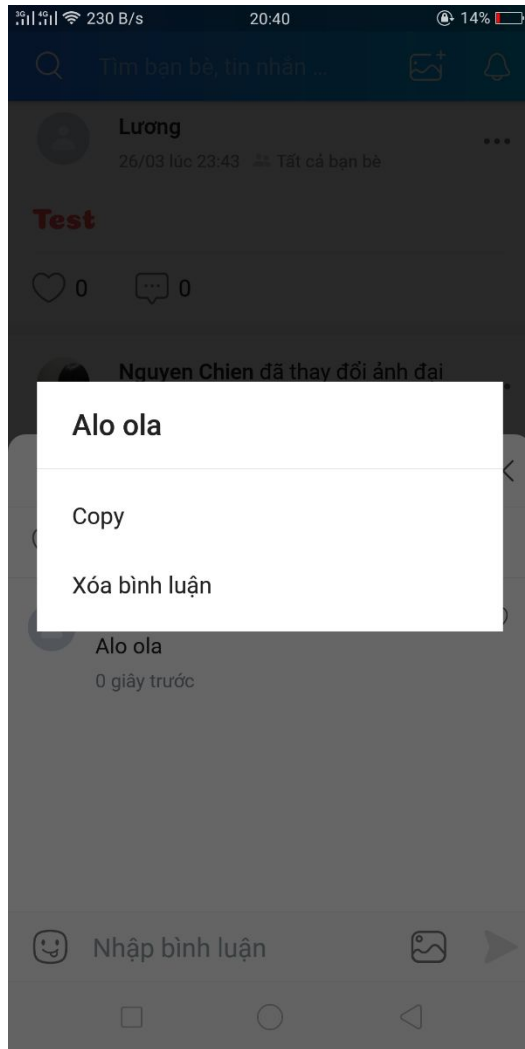
Các test case cho del_comment (4)

6. 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**”.
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

Mục lục

1. API Báo cáo bài viết
2. Giao diện báo cáo bài viết
3. API Lấy danh sách bình luận
4. Giao diện danh sách bình luận bài viết
5. API Bình luận bài viết
6. Giao diện bình luận bài viết
7. API xóa bài viết
8. **Giao diện xóa bình luận**
9. API chỉnh sửa bình luận
10. Giao diện chỉnh sửa bình luận
11. API thích bài viết

Giao diện xóa bình luận



Giao diện này được xuất hiện sau khi người dùng ấn và giữ vào bình luận sau 1 khoảng thời gian.

Tại đây giao diện sẽ hiển thị bình luận và đưa ra chức năng xóa bình luận cho người sử dụng.

Sau khi bạn ấn xóa bình luận thì sẽ xóa luôn không có bảng hiển thị xác nhận xóa.

Mục lục

1. API báo cáo bài viết
2. Giao diện báo cáo bài viết
3. API lấy danh sách bình luận
4. Giao diện danh sách bình luận bài viết
5. API bình luận bài viết
6. Giao diện bình luận bài viết
7. API xóa bài viết
8. Giao diện xóa bài viết
9. **API chỉnh sửa bình luận**
10. Giao diện chỉnh sửa bình luận
11. API thích bài viết

edit_comment

Input

NO	Tên parameter	Type	NN	Mô tả
1	token	string	0	
2	id	string	0	id của bài viết
3	id_com	string	0	id của bình luận
4	comment	string	0	nội dung đã chỉnh sửa

Output

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

Mô tả edit_comment

API thực hiện việc sửa **một bình luận** trong một bài viết của tài khoản bình luận.

Request dạng: **POST**.

Tham số: **token**, **id** (của bài viết), **id_com**(của bình luận), **comment** (nội dung đã chỉnh sửa).

Kết quả đầu ra: 1000|OK - Nếu thành công thì sẽ cập nhật lại bài viết.

Nếu không thành công thì sẽ thông báo lỗi tương ứng.

Các test case cho edit_comment (1)

1. Người dùng truyền đúng mã phiên đăng nhập, id bài viết và các tham số khác phù hợp.
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, toàn bộ phần nội dung đang chỉnh sửa sẽ bị xóa đi ở phía client.

Các test case cho edit_comment (2)

3. Người dùng truyền đúng mã phiên đăng nhập, id bài viết và các tham số khác phù hợp nhưng nội dung không phù hợp để đăng (ví dụ ảnh giết hại động vật).
Kết quả mong đợi: mã lỗi 1010, ứng dụng hiển thị ra các thông tin cần thiết.
4. Người dùng truyền đúng mã phiên đăng nhập, id bài viết và các tham số khác phù hợp nhưng tài khoản bị khóa.
Kết quả mong đợi: mã lỗi 9995, ứng dụng hiển thị ra các thông tin cần thiết và đẩy người dùng sang trang đăng nhập

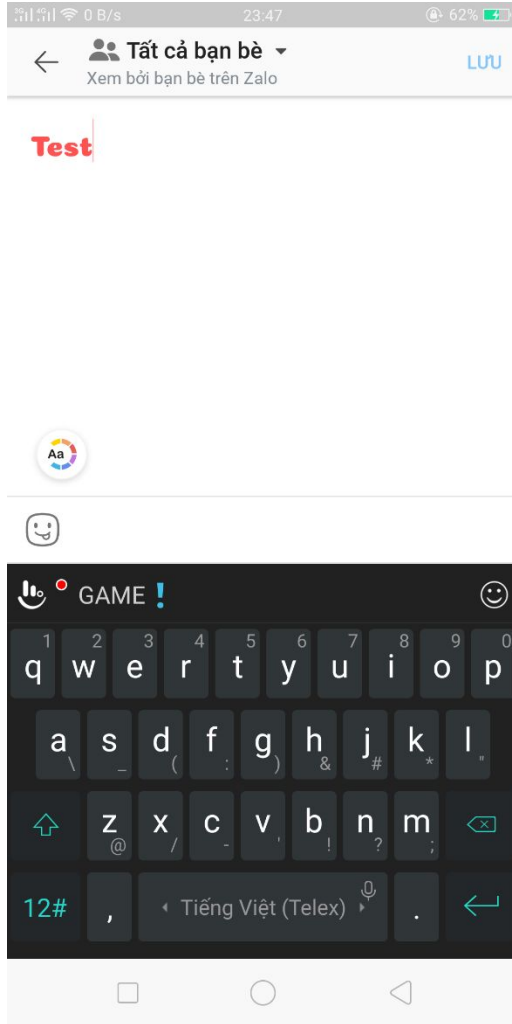
Các test case cho edit_comment (3)

5. Người dùng truyền đúng mã phiên đăng nhập, id và các tham số khác phù hợp nhưng tài khoản bị block.
Kết quả mong đợi: Thông báo tài khoản đã bị chặn.
6. Người dùng truyền đúng mã phiên đăng nhập, id bài viết và các tham số khác phù hợp nhưng tài khoản bị khóa.
Kết quả mong đợi: mã lỗi 9995, ứng dụng hiển thị ra các thông tin cần thiết và đẩy người dùng sang trang đăng nhập.

Mục lục

1. API báo cáo bài viết
2. Giao diện báo cáo bài viết
3. API lấy danh sách bình luận
4. Giao diện danh sách bình luận bài viết
5. API bình luận bài viết
6. Giao diện bình luận bài viết
7. API xóa bài viết
8. Giao diện xóa bài viết
9. API chỉnh sửa bình luận
- 10. Giao diện chỉnh sửa bình luận**
11. API thích bài viết

Giao diện edit_comment



Giống 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

Mục lục

1. API báo cáo bài viết
2. Giao diện báo cáo bài viết
3. API lấy danh sách bình luận
4. Giao diện danh sách bình luận bài viết
5. API bình luận bài viết
6. Giao diện bình luận bài viết
7. API xóa bài viết
8. Giao diện xóa bài viết
9. API chỉnh sửa bình luận
10. Giao diện chỉnh sửa bình luận
11. **API thích bài viết**

like

Input

NO	Tên parameter	Type	NN	Mô tả
1	token	string	0	
2	id	string	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: 1000|OK - 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)

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.

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).

Các test case cho like (3)

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.

Các test case cho like (4)

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 (5)

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 (6)

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).

