

BÀI TẬP TUẦN 5

Nhắc lại

Đã nắm được tất cả các API và giao diện liên quan đến bình luận

Làm việc với API `get_list_post`, `check_new_items` và màn hình trang chủ

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. API danh sách các bài
2. Giao diện trang chủ
3. API kiểm tra phần tử mới

MỤC LỤC

1. **API danh sách các bài**
2. Giao diện trang chủ
3. API kiểm tra phần tử mới

get_list_posts

Tên parameter	Type	NN	Mô tả
token	string	X	
user_id	string	X	id người dùng
in_campaign	string	X	0: lấy tất cả sp, 1: chỉ lấy sp trong chiến dịch KM
campaign_id	string	X	id của chiến dịch quảng cáo, dùng cho lấy sản phẩm trong campaign
latitude	string	X	tọa độ latitude của người dùng
longitude	string	X	tọa độ longitude của người dùng
last_id	string	X	last_id trả về lần trước
index	string	O	index start select
count	string	O	số bài trong 1 lần gửi yêu cầu

get_list_posts (2)

Tên parameter	Type	NN	Mô tả
code	string	0	(tham khảo phần response common)
message	string	0	
data	array	0	
posts	array	0	
id	string	0	
name	string	0	
image	array	X	array url hình ảnh
video	array	X	
url	string	X	nếu không có video thì bằng trống
thumb	string	X	nếu không có video thì bằng trống
described	string	0	
created	string	0	thời gian đăng
like	string	0	số like của sản phẩm
comment	string	0	số comment của sản phẩm
is_liked	string	0	kiểm tra User đã like chưa. 1: đã like, 0: chưa like
is_blocked	string	0	

get_list_posts (3)

	can_comment	string	0	
	can_edit	string	0	
	banned	string	0	
	state	string	0	
	author	string	0	
	id	string	0	
	username	string	0	
	avatar	string	0	
	online	string	X	trạng thái online
	new_items	string	0	số bài viết mới đăng
	last_id	string	0	id bài viết cuối
in_campaign		string	0	cập nhật in_campaign
campaign_id		string	0	cập nhật id của campaign

Mô tả get_list_posts

API thực hiện việc lấy danh sách các bài viết để hiện lên trang chủ cho một tài khoản của người dùng

Request dạng POST

Tham số: mã **token** phiên đăng nhập, **id** của người dùng, **in_campaign** (chỉ lấy sản phẩm khuyến mãi hay không), chiến dịch quảng cáo, tọa độ người dùng, **last_id** của bài viết mới nhất, chỉ số **index** và **count**

Đầu vào in_campaign: liên quan đến việc có lấy các sản phẩm khuyến mại của các chiến dịch quảng cáo hay không?

Mô tả get_list_posts (2)

Chiến dịch quảng cáo được đăng ký bởi các công ty, các công ty này đăng bài để bán hàng và liên lạc với bên quản trị server để báo về các chiến dịch quảng cáo cũng như các sản phẩm khuyến mãi.

Nhóm SV (cụ thể là các bạn làm Web) cần xây dựng chức năng cho phép công ty đăng kí các chiến dịch quảng cáo và sản phẩm khuyến mãi.

Đầu vào campaign_id là lưu trữ định danh của chiến dịch quảng cáo cần hiển thị. Chú ý rằng hai đầu vào liên quan đến quảng cáo sẽ được cập nhật liên tục khi gọi API này.

Mô tả get_list_posts (3)

Tọa độ người dùng: dùng để hạn chế các bài viết không phù hợp với quốc gia hoặc hiển thị quảng cáo theo địa phương tốt hơn.

Server sẽ lưu trữ các thông số này để phục vụ tốt hơn.

Đầu vào last_id để phục vụ việc hiển thị các bài viết mới chính xác hơn. Tại sao?

Tình huống lúc 9:00AM, ứng dụng nhận được dữ liệu bài viết từ A1 đến A20. Lúc 9:01AM, server có bài viết mới B1 đến B10. Lúc 9:02AM, ứng dụng yêu cầu thêm 20 bài nữa (index = 20 và count = 20).

Vậy server sẽ trả về bài A21 đến A40 hay bài A11 đến A30? Làm thế nào để ứng dụng biết còn có các bài mới B1 đến B10?

Mô tả get_list_posts (4)

Ứng dụng không bao giờ yêu cầu lấy hết dữ liệu mà sẽ chỉ lấy theo từng phần.

Mặc định từ phía ứng dụng, tất cả các yêu cầu gửi đến server có giá trị chỉ số thì count luôn bằng 20.

Tuy vậy phía server luôn tâm niệm rằng giá trị này có thể thay đổi.

Kết quả đầu ra: nếu lấy dữ liệu thành công thì sẽ có gửi đầy đủ các thông tin cần thiết cho ứng dụng. Nếu không thành công thì sẽ có các thông báo lỗi tương ứng

Mô tả `get_list_posts` (5)

Dữ liệu trả về trong trường `data` (nếu thành công) là mảng các bài viết

- Có id của bài viết, ảnh bài viết (nếu có) hoặc video của bài viết (nếu có)

- Có name lưu trữ danh sách những người bạn có tương tác với bài viết (thích/bình luận)

- Có nội dung của bài viết trong trường `described`, thời gian đăng bài viết

- Có số lượt thích bài viết, có số bình luận của bài viết

- Có trường `can_comment` để xem bài viết đó có được bình luận hay không?

- Có trường `can_edit` để bài viết đó có được chỉnh sửa hay không (nếu đang bị khóa sửa bài thì không sửa được)

Mô tả get_list_posts (6)

Dữ liệu trả về trong trường data (nếu thành công) là mảng các bài viết (tiếp)

Banned: báo rằng bài viết bị che đi tất cả các ảnh/video hoặc một số ảnh

State: Trạng thái cảm xúc của bài viết

Author: trường thông tin liên quan đến tác giả bài viết, có đầy đủ id, tên, ảnh đại diện và trạng thái online

New_items: số lượng các bài viết mới đăng bị bỏ lỡ

Last_id: cập nhật tham số để gửi lên lần sau (nếu lấy thêm bài)

Các in_campaign, campaign_id: cập nhật hai tham số này

Mô tả get_list_posts (6)

Dữ liệu trả về trong trường data (nếu thành công) là mảng các bài viết (tiếp)

Banned: báo rằng bài viết bị che đi tất cả các ảnh/video hoặc một số ảnh

State: Trạng thái cảm xúc của bài viết

Author: trường thông tin liên quan đến tác giả bài viết, có đầy đủ id, tên, ảnh đại diện và trạng thái online

CÂU HỎI 1: TẠI SAO LẠI CẦN THAM SỐ LAST_ID CỦA SERVER TRẢ VỀ? ĐÁNG LẼ CÓ THỂ LẤY ID CỦA BÀI VIẾT CUỐI CÓ TRONG DANH SÁCH?

Các test case cho get_list_posts

1. Người dùng truyền đúng mã phiên đăng nhập, các tham số khác cũng đầy đủ và đúng đắn

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 get_list_posts (2)

3. Người dùng truyền đúng mã phiên đăng nhập, và các tham số khác. Nhưng không còn bài viết nào

Kết quả mong đợi: mã lỗi báo không còn dữ liệu nữa, nhưng ứng dụng không hiển thị thông báo lỗi này, chỉ đơn giản thông báo người dùng có thể kết bạn thêm.

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

Kết quả mong đợi: người dùng bị đăng xuất, ứng dụng sẽ phải đẩy người dùng sang trang đăng nhập, tất cả các dữ liệu lưu trữ trước nếu có đều bị xóa.

Các test case cho get_list_posts (3)

5. Người dùng truyền đúng mã phiên đăng nhập, id bài viết. Nhưng dữ liệu trả về có trường nội dung bài viết bị lỗi

Kết quả mong đợi: ứng dụng không hiển thị những bài viết bị lỗi (sai định dạng khiến không chuyên đổi). Nếu tất cả bài mới lấy về bị lỗi thì ứng dụng hiển thị như thể không còn bài viết mới nào nữa.

6. Người dùng truyền đúng mã phiên đăng nhập, id bài viết. Nhưng dữ liệu trả về có trường like hoặc comment hoặc is_liked bị lỗi.

Kết quả mong đợi: ứng dụng sẽ phải căn cứ dữ liệu trả về mà hiển thị được các thông tin cần thiết. Các trường bị lỗi coi như nhận giá trị mặc định (bằng 0)

Các test case cho get_list_posts (4)

7. Người dùng truyền đúng mã phiên đăng nhập, và các tham số khác. Nhưng dữ liệu trả về có trường can_comment thông báo bình luận bị khóa hoặc trường can_comment bị lỗi về giá trị

Kết quả mong đợi: ứng dụng không hiển thị ô để nhập bình luận.

8. Người dùng truyền đúng mã phiên đăng nhập, và các tham số khác. Nhưng dữ liệu trả về có trường author bị lỗi về id.

Kết quả mong đợi: ứng dụng không hiển thị bài viết có thông tin tác giả bị lỗi về id. Các trường khác của author bị lỗi thì vẫn có thể nhận giá trị mặc định để hiển thị bài viết

Các test case cho get_list_post (5)

9. Người dùng truyền đúng mã phiên đăng nhập, và các tham số khác. Nhưng dữ liệu trả về có trường described và trường đa phương tiện (image hoặc video) bị lỗi

Kết quả mong đợi: không hiển thị bài viết.

10. Người dùng truyền đúng mã phiên đăng nhập và các tham số khác. Nhưng dữ liệu trả về có một trong hai trường described và trường đa phương tiện (image hoặc video) bị lỗi.

Kết quả mong đợi: ứng dụng vẫn hiển thị bài viết cho các trường thông tin không bị lỗi. Các trường bị lỗi thì vẫn có thể nhận giá trị mặc định để hiển thị bài viết

Các test case cho get_list_post (6)

11. Người dùng truyền đúng mã phiên đăng nhập, và các tham số khác. Có bài viết cần phải che đi hình ảnh/video không thích hợp nhưng ứng dụng bị lỗi trong việc hiển thị màn che

Kết quả mong đợi: phải bắt được sự kiện lỗi hiển thị màn che để cuối cùng là xóa đi bài viết trong danh sách.

12. Người dùng truyền đúng mã phiên đăng nhập và các tham số khác. Nhưng dữ liệu trả về có từ một trong hai trường in_campaign và campaign_id bị lỗi.

Kết quả mong đợi: ứng dụng vẫn hiển thị bài viết cho các trường thông tin không bị lỗi. Các trường bị lỗi thì vẫn có thể nhận giá trị mặc định (hoặc giá trị cũ) để tiếp tục gửi yêu cầu

Các test case cho get_list_post (7)

13. Người dùng truyền đúng mã phiên đăng nhập, và các tham số khác nhưng tham số last_id bị lỗi (không có hoặc không đúng định dạng).

Kết quả mong đợi: mã lỗi sai giá trị dữ liệu của tham số. Ứng dụng không nhận được thêm danh sách nào cả.

14. Người dùng truyền đúng mã phiên đăng nhập, và các tham số khác nhưng tham số index hoặc count bị lỗi (không có hoặc không đúng định dạng).

Kết quả mong đợi: mã lỗi sai giá trị dữ liệu của tham số. Ứng dụng không nhận được thêm danh sách nào cả.

Các test case cho get_list_post (8)

15. Người dùng truyền đúng mã phiên đăng nhập, và các tham số khác nhưng các tham số tọa độ bị lỗi (không có hoặc không đúng định dạng).

Kết quả mong đợi: server sử dụng lại tham số lần gần nhất.

16. Người dùng truyền đúng mã phiên đăng nhập, và các tham số khác nhưng tham số user_id của người dùng bị lỗi.

Kết quả mong đợi: ứng dụng sẽ phải đẩy người dùng sang trang đăng nhập.

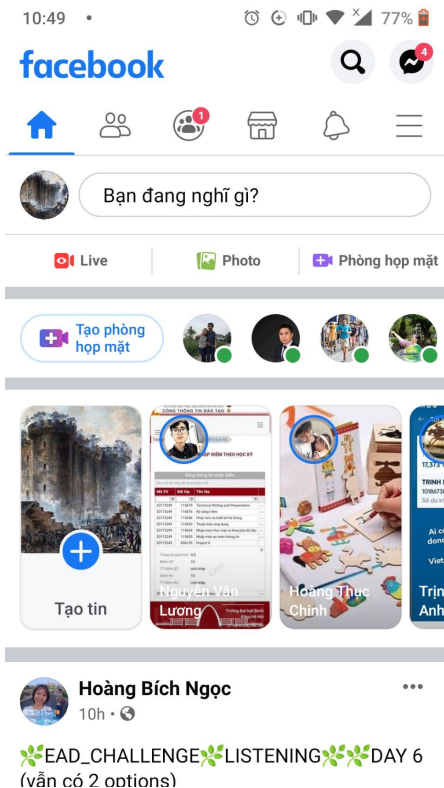
Các test case cho get_list_post (9)

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

MỤC LỤC

1. API danh sách các bài
2. **Giao diện trang chủ**
3. API kiểm tra phần tử mới

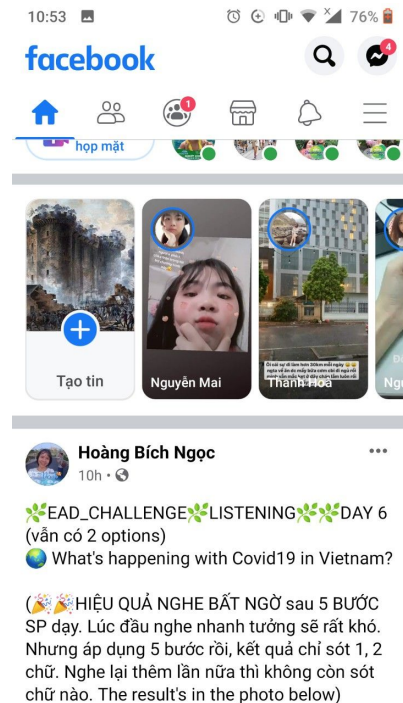
Giao diện trang chủ



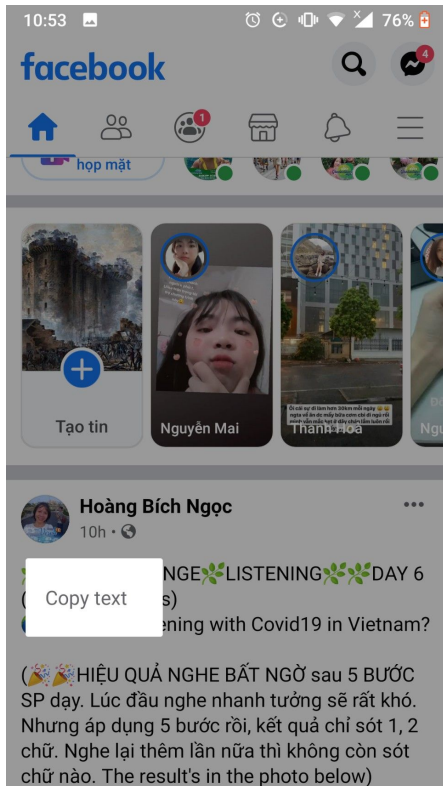
So với ứng dụng Facebook, ứng dụng này sẽ được giảm bớt các chức năng sau:

- + Hàng “Live/Photo/Phòng họp mặt”
- + Hàng “Tạo phòng họp mặt”
- + Hàng “Tạo tin”

Để ý các hiệu ứng vuốt lên vuốt xuống sẽ khiến các hàng trên cùng được di chuyển cùng (chức năng này là tiêu chuẩn ở mọi ứng dụng)



Giao diện trang chủ (2)



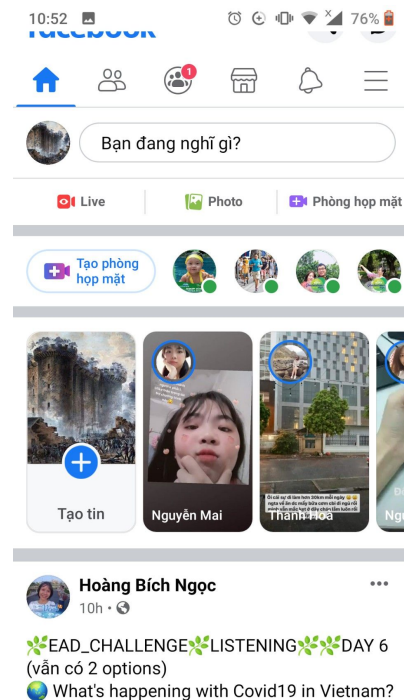
Ứng dụng cho phép người dùng sao chép được bài viết.

Trong danh sách các tab của ứng dụng, có tab thứ nhất, thứ 2 giống Facebook

Tab thứ 3 là liệt kê danh sách các video được thu thập

Tab thứ 4 là tab thông báo

Tab thứ 5 giống với tab thứ 6 của Facebook



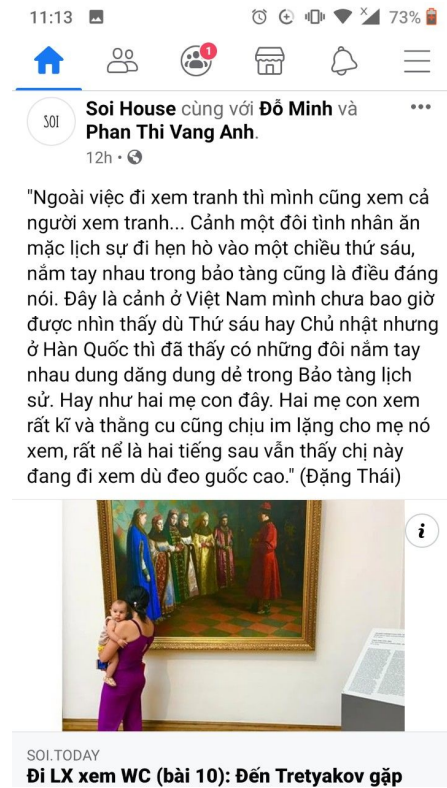
Giao diện trang chủ (3)



Với các bài viết quá dài và có “Xem thêm” (hoặc See more) thì khi nhấn vào sẽ mở rộng ra

Có thể làm chức năng nhấn thêm lần nữa (vào bài viết) để thu gọn hoặc nút “Thu gọn”

Để đơn giản, ứng dụng sẽ mở rộng trong trang hiện tại nếu bài viết chỉ có một bức ảnh và dưới 100 lượt thích và dưới 100 bình luận



Giao diện trang chủ (4)



Trong trường hợp còn lại, bài viết có nhiều bức ảnh và lượt thích cùng bình luận thì ứng dụng sẽ chuyển sang một trang riêng để hiển thị

Nhấn nút Back trên máy hoặc trên ứng dụng để quay lại màn hình trước đó.

Chú ý vẫn giữ lại nút “Tìm kiếm” như ở góc trên cùng bên phải của ảnh bên phải



Giao diện trang chủ (5)



Những bài viết tiếng Anh không cần phải hiển thị chức năng dịch

Sẽ hiển thị những người bạn có tương tác với những bài hiển thị trên trang chủ.

Trong trường hợp có nhiều người có tương tác thì chỉ hiện tối đa tên của hai người thôi.

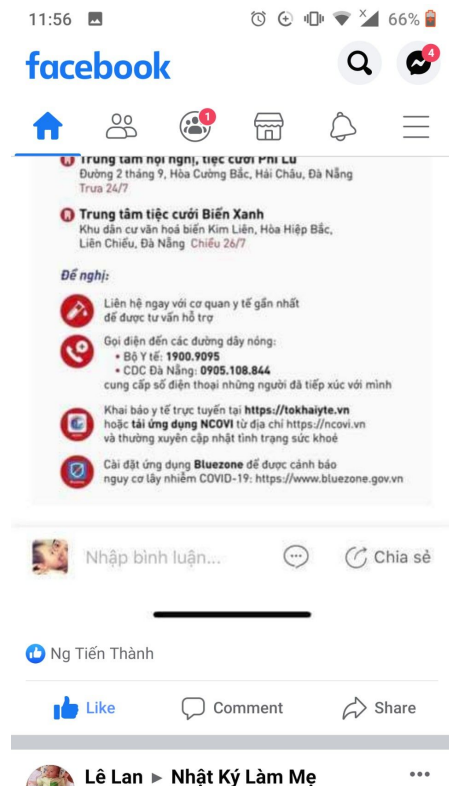


Giao diện trang chủ (6)



Những bài còn rất mới (được đăng dưới 10 phút trước đây) và người viết bài còn online thì hiển thị màu của thời gian là xanh da trời.

Khi chỉ có mỗi người chủ ứng dụng tiến hành thao tác thích bài viết thì ứng dụng có hiển thị như hình bên



Giao diện trang chủ (7)



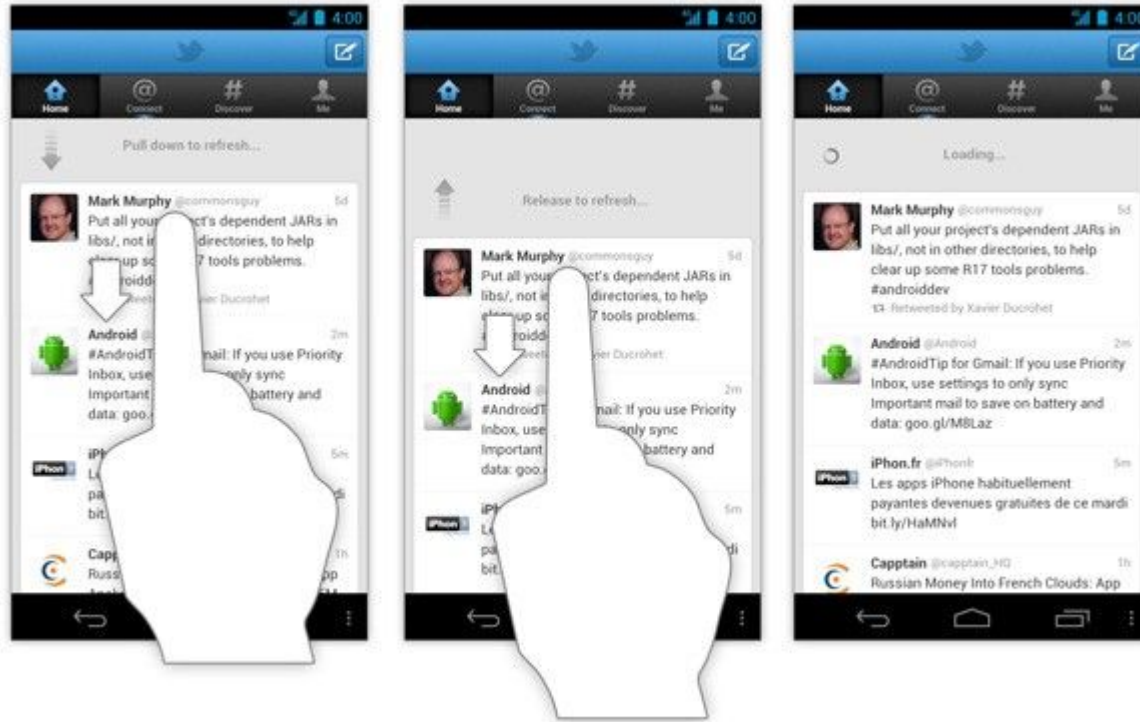
Ứng dụng cho phép vuốt sang bên trái hoặc phải để chuyển tab

Hoặc nhấn vào các biểu tượng để chuyển qua lại các tab

Chú ý việc chuyển qua lại như thế phải có chuyển động (animation) chứ không được đổi giao diện ngay lập tức.



Giao diện trang chủ (8)



Cần tạo trải nghiệm pull down và pull up để làm mới dữ liệu và nạp thêm dữ liệu mới

Khi pull down thành công thì làm mới toàn bộ danh sách

Khi pull up thì sẽ nạp thêm vào danh sách phần tử

Giao diện trang chủ (9)

Pull up to load more: là trải nghiệm tải sẵn cho người dùng khi người dùng cuộn gần đến những phần tử cuối cùng trong danh sách

Khi đó, với những phần tử giao diện được hiện thực hóa trên màn hình thì sẽ nảy sinh ra một sự kiện

Viết mã nguồn bắt sự kiện đó, kiểm tra xem phần tử được hiển thị có phải gần đến một ngưỡng (threshold) nào đó và nếu đúng thì gửi yêu cầu lên Server để lấy dữ liệu về.

Nếu các thao tác pull down và pull up gặp tình huống không kết nối mạng thì ứng dụng vẫn báo cho người dùng nhưng không thay đổi giao diện (trừ khi không có dữ liệu nào trong cache)

Giao diện trang chủ (10)

Một cách nữa để tạo trải nghiệm cho người dùng rằng ứng dụng nạp dữ liệu rất nhanh cho màn hình trang chủ:

- + Đó là dùng cache để lưu trữ lại kết quả của lần truy vấn trước.
- + Cache này chỉ áp dụng cho các tab của màn trang chủ
- + Cache này sẽ trống nếu người dùng login lần đầu vào ứng dụng hoặc lần trước đã đăng xuất (hoặc bị đăng xuất)
- + Cache này sẽ càng ngày càng nhiều nếu người dùng tiếp tục thực hiện thao tác pull up to load more
- + Cache sẽ được làm mới lại nếu người dùng thực hiện pull down to refresh

Giao diện trang chủ (11)

Đề tạo trải nghiệm cho người dùng rằng ứng dụng nạp dữ liệu rất nhanh cho màn trang chủ (tiếp):

- + Khi vào một tab nào của màn trang chủ, nếu đã có cache thì không gửi yêu cầu `get_list_post` (hoặc các api khác của các tab). Ngoại trừ tab liên quan đến push và tab cài đặt.
- + Nếu đã có cache thì gửi yêu cầu **`check_new_item`** để kiểm tra các phần tử mới chưa được hiện trong tab.
- + Nếu một lý do cache bị lỗi không chuyển đổi từ JSON sang đối tượng được thì xóa đi và gửi yêu cầu nạp dữ liệu
- + Có lẽ không nên dùng CSDL để lưu cache này.

Giao diện trang chủ (12)

Đề tạo trải nghiệm cho người dùng rằng ứng dụng nạp dữ liệu rất nhanh cho màn trang chủ (tiếp):

- + Với cache, người dùng sẽ thấy ứng dụng hiển thị được gần như ngay lập tức dữ liệu bài viết
- + Cache giúp ứng dụng vẫn hiển thị được dữ liệu ngay khi không có mạng
- + Khi người dùng có thao tác thích hoặc bỏ thích một bài nào đó, cũng sẽ cập nhật ngay vào cache (nếu server hồi đáp thành công).
- + Nếu nhấn thích hoặc bỏ thích khi không có kết nối Internet thì ứng dụng KHÔNG hồi đáp mà báo không có kết nối Internet (khác với Facebook).

Giao diện trang chủ (13)



Để tạo trải nghiệm cho người dùng rằng ứng dụng nạp dữ liệu rất nhanh cho màn trang chủ (tiếp):

- + Không bắt buộc ứng dụng phải lưu trữ trong cache cả danh sách bình luận của một bài viết

Trong trang chủ vẫn cho phép hiển thị nút “Tìm kiếm” và nút bật cửa sổ chat.

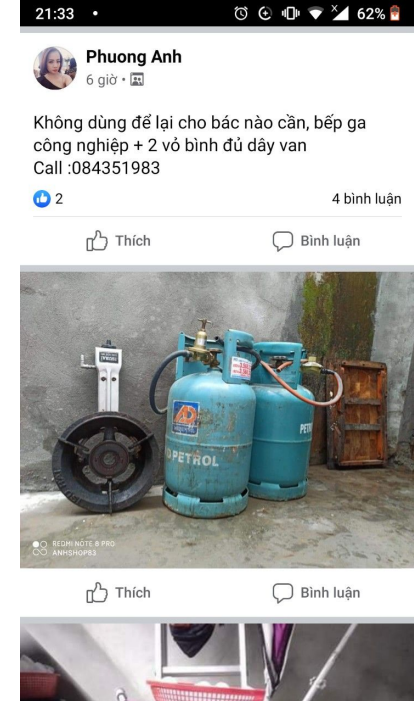
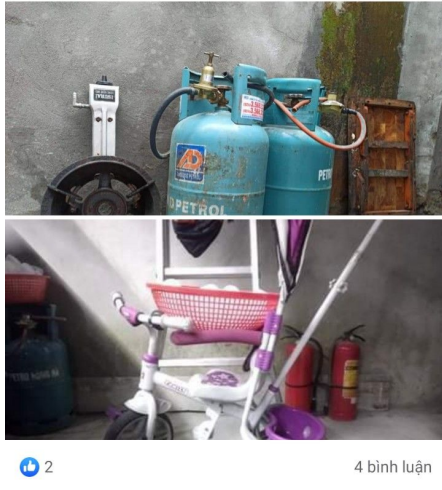
Sinh viên có thể tùy chọn làm cửa sổ chat là một popup hay một ứng dụng khác, độc lập.

Giao diện trang chủ (14)



Khi nhấp chuột vào từng bức ảnh của một bài viết trong danh sách, ứng dụng sẽ chuyển sang “Trang hiển thị bài viết” để hiển thị lại các ảnh theo danh sách dạng List View

Nếu mạng không có kết nối, ứng dụng có thể hiển thị lại dữ liệu trong cache.



Giao diện trang chủ (15)



Với các video, nếu nhóm làm được, sẽ có biểu tượng play như hình bên trái

Khi xem video, do bài viết ở màn trang chủ nên ứng dụng sẽ chuyển sang khu vực **Watch**

Chú ý rằng dữ liệu của **Watch** sẽ có api **get_list_videos** để phục vụ hiển thị



Giao diện trang chủ (16)

Khi người dùng đăng một bài mới trong màn trang chủ, nếu đăng thành công thì bài đó sẽ được xuất hiện ở phần tử đầu tiên trong danh sách bài viết của trang chủ

- + Ứng dụng tự động thêm vào.

Vậy **last_id** sẽ được thay đổi? Không, trường **last_id** vẫn là kết quả do server trả về từ trước.

MỤC LỤC

1. API danh sách các bài
2. Giao diện trang chủ
3. **API kiểm tra phần tử mới**

check_new_item

Input

NO	Tên parameter	Type	NN	Mô tả
	last_id	string	0	
	category_id	string	X	

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	
	new_items	string	0	

Mô tả check_new_item

API thực hiện việc kiểm tra xem một tab của trang chủ có phần tử mới không?

Request dạng POST

Tham số: **last_id** và **category_id** (không bắt buộc có giá trị ban đầu bằng 0),

Kết quả đầu ra: nếu thành công thì sẽ được nhận số phần tử mới của Tab chưa được hiển thị. Nếu không thành công thì sẽ có các thông báo lỗi tương ứng

Trường category_id có giá trị từ 0 đến 3. Nếu không truyền thì mặc định sẽ là 0.

Các test case cho check_new_item

1. Người dùng truyền đúng mã last_id, category_id đúng định dạng

Kết quả mong đợi: 1000 | OK (Thông báo thành công), gửi cho ứng dụng số lượng phần tử mới của tab. Ứng dụng tự hiển thị một nút nổi ngay trên phần đầu tiên của danh sách bài viết

2. Người dùng gửi sai một trong hai mã last_id, category_id (không truyền hoặc giá trị không hợp lệ)

Kết quả mong đợi: ứng dụng sẽ kiểm tra trước để không gửi dữ liệu không hợp lệ lên. Tuy vậy nếu lỡ gửi lên thì khi hệ thống trả về lỗi, ứng dụng sẽ không cần hiển thị thông báo cho người dùng bình thường biết.

Các test case cho check_new_item

1. Người dùng truyền đúng mã last_id, category_id đúng định dạng

Kết quả mong đợi: 1000 | OK (Thông báo thành công), gửi cho ứng dụng số lượng phần tử mới của tab. Ứng dụng tự hiển thị một nút nổi ngay trên phần đầu tiên của danh sách bài viết

2. **CHÚ Ý: KHÔNG HIỂN THỊ CHO NGƯỜI DÙNG BÌNH THƯỜNG NHƯNG VẪN HIỂN THỊ KHI TRONG GIAI ĐOẠN PHÁT TRIỂN.**

Kết quả mong đợi: ứng dụng sẽ kiểm tra trước để không gửi dữ liệu không hợp lệ lên. Tuy vậy nếu lỡ gửi lên thì khi hệ thống trả về lỗi, ứng dụng sẽ không cần hiển thị thông báo cho người dùng bình thường biết.

Các test case cho check_new_item (2)

3. Người dùng truyền các tham số hợp lệ nhưng giá trị new_items trả về không hợp lệ.

Kết quả mong đợi: Ứng dụng coi như giá trị new_items bằng 0.

4. Người dùng truyền các dữ liệu tham số hợp lệ và new_items trả về hợp lệ nhưng người dùng không còn ở trong giao diện trang chủ nữa.

Kết quả mong đợi: Ứng dụng lưu new_items vào trong cache, nhưng kể cả khi quay lại màn hình trang chủ, thì vẫn gửi yêu cầu check_new_item.

Các test case cho check_new_item (3)

5. Người dùng truyền các dữ liệu tham số hợp lệ và new_items trả về hợp lệ nhưng người dùng đã bị đẩy sang trang đăng nhập.

Kết quả mong đợi: Ứng dụng bỏ qua không xử lý dữ liệu được trả về này.

6. Ứng dụng đã truyền đủ các dữ liệu tham số hợp lệ nhưng server bị lỗi không lấy được giá trị new_items này.

Kết quả mong đợi: mã lỗi trả về, ứng dụng vẫn ở nguyên màn hình trang chủ.

Các test case cho check_new_item (4)

7. Ứng dụng đã truyền đủ các dữ liệu tham số hợp lệ nhưng trong quá trình kiểm tra thì mạng bị ngắt kết nối.

Kết quả mong đợi: ứng dụng cần kiểm tra ngay phía mình để đảm bảo đưa ra thông báo về kết nối mạng bị ngắt.

HẾT TUẦN 5