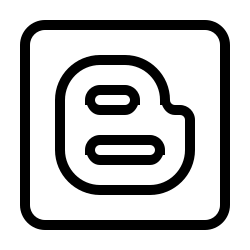
TRƯỜNG ĐẠI HỌC

CÔNG NGHIỆP THỰC PHẨM TP HCM

**BÀI BÁO CÁO**

Đề tài:

Xây dựng app di động lấy dữ liệu từ website dạng blogger



**Thành viên:**

**Nguyễn Quốc Việt 2001170234**

**Bùi Vũ Trường 2001170208**

**Nguyễn Thị Ngọc Thu 2001170439**

Giáo Viên Hướng Dẫn: Vũ Văn Vinh

Thành Phố Hồ Chí Minh, Tháng 07, Năm 2020

**MỤC LỤC**

[phần 1: giới thiệu đề tài 1](#_Toc45175177)

[phần 2: các công cụ cần thiết 2](#_Toc45175178)

[phần 3: các chức năng của ứng dụng 3](#_Toc45175179)

[phần 4: bảng tổng hợp các link API dùng trong ứng dụng 5](#_Toc45175180)

[I. website và database 5](#_Toc45175181)

[1 Bảng dữ liệu user\_like\_post 6](#_Toc45175182)

[2 Bảng dữ liệu wp\_comment 6](#_Toc45175183)

[3 Bảng dữ liệu wp\_post 7](#_Toc45175184)

[4 Bảng dữ liệu wp\_terms 7](#_Toc45175185)

[5 Bảng dữ liệu wp\_term\_relationships 8](#_Toc45175186)

[6 Bảng dữ liệu wp\_users 8](#_Toc45175187)

[II. API 9](#_Toc45175188)

[phần 5: bảng phân công đánh giá cho từng thành viên 12](#_Toc45175189)

[phần 6: Nội dung và các tính năng trong ứng dụng 13](#_Toc45175190)

[I. các kiến thức mới cần biết 13](#_Toc45175191)

[1 tạo server ảo cho web php bằng ứng dụng xampp ? 13](#_Toc45175192)

[2 tạo database và tạo api ? 13](#_Toc45175193)

[3 gọi api trên mạng cục bộ ? 16](#_Toc45175194)

[4 tăng tốc ứng dụng bằng cách làm việc trên luồng phụ, cập nhật giao diện lên luồng chính ? 17](#_Toc45175195)

[II. Mô phỏng tất cả quy trình các activity hoạt động 19](#_Toc45175196)

[1 Màn hình chờ (loading activity) 19](#_Toc45175197)

[2 giao diện màn hình chính (main activity) 20](#_Toc45175198)

[3 giao diện cài đặt (setting Activity) 22](#_Toc45175199)

[4 giao diện đăng bài mới 22](#_Toc45175200)

[5 giao diện thông tin cá nhân 23](#_Toc45175201)

[6 đăng nhập và quên mật khẩu. 24](#_Toc45175202)

[7 đăng ký và đăng ký nhanh 25](#_Toc45175203)

[III. tìm hiểu từng tính năng trong ứng dụng 26](#_Toc45175204)

[1 load danh mục bài viết lên menu 26](#_Toc45175205)

[2 Load tất cả biết viết, hay lọc bài viết theo từng danh mục 26](#_Toc45175206)

[3 Tạo tài khoản người dùng, đăng nhập, đăng xuất 26](#_Toc45175207)

[4 Tìm kiếm, lọc các bài viết 26](#_Toc45175208)

[5 Đa ngôn ngữ 26](#_Toc45175209)

[6 Xem thông tin chi tiết bài viết, like, bình luận 26](#_Toc45175210)

[7 Đăng bài viết mới 26](#_Toc45175211)

[8 Quản lý cá nhân cho tài khoản đã đăng nhập 26](#_Toc45175212)

[9 Gởi Email khôi phục tài khoản 26](#_Toc45175213)

[10 thông báo bài viết mới 26](#_Toc45175214)

[11 Chỉnh màu theme tối hoặc sáng cho ứng dụng và sử dụng các hiệu ứng Animation 26](#_Toc45175215)

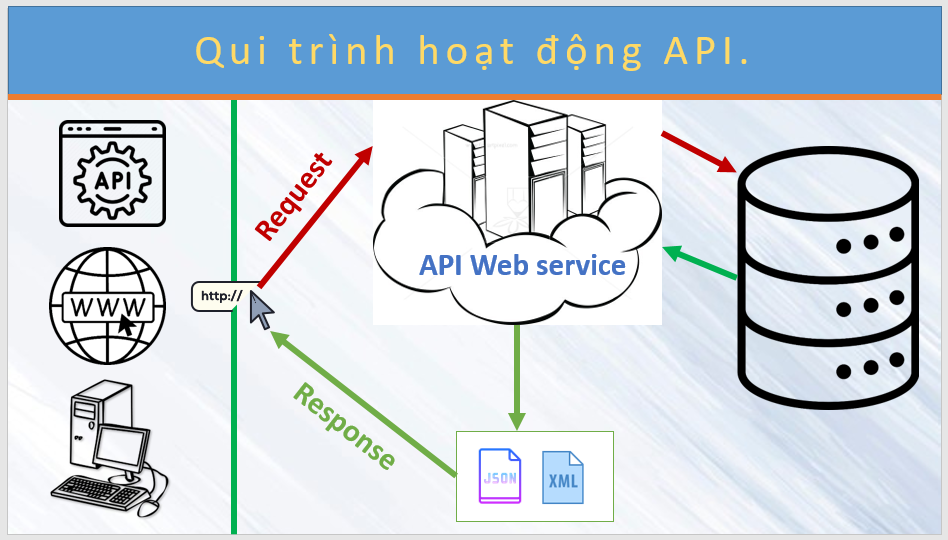
[12 Tìm kiếm bài viết bằng giọng nói 26](#_Toc45175216)

phần 1: giới thiệu đề tài

Ngày nay với sự phát triển mạnh mẽ của khoa học và công nghệ thì điện thoại di động ngày càng trở nên phổ biến với người sử dụng hơn bao giờ hết. Đặt biệt là với thiết kế nhỏ gọn, phù hợp với mọi lứa tuổi. Để được tiếp cận với khoa học và công nghệ thì một điều mà con người luôn luôn được quan tâm đến đầu tiên đó là nhất điện thoại lên và xem tin tức. Tin tức có thể bao gồm nhiều thể loại khác nhau, tùy thuộc vào những sở thích những độ tuổi khác nhau, bao gồm cả tin tin chính trị, tin về khoa học, tin về dạng học tập,... Nhưng suy cho cùng, tin tức luôn là một khía cạnh vô cùng quan trọng và xoay quanh đời sống của con người.

Nắm bắt được tầm nhìn và sự quan trọng ấy, cùng với kiến thức học tại trường. Nhóm em gồm ba thành viên (Nguyễn Quốc Việt, Bùi Vũ Trường, Nguyễn Thị Ngọc Thu) đã quyết định chọn đề tài: xây dựng app di động lấy dữ liệu từ website dạng blogger(cung cấp các tin tức mới).

phần 2: các công cụ cần thiết

1. Công cụ lập trình
   1. Android studio
   2. Máy ảo, hoặc máy thật android dùng để test chương trình
2. Dữ liệu
   1. Website dạng blogger: website này có tên miền “myHufier.ga”. Hoặc có thể dùng Localhost, dùng ứng dụng Xampp để giả lập Server. Lưu ý website chạy trên mã nguồn mở của WordPress. Có thể download source code từ thư mục gốc về bằng link: [rootfoldder](https://google.com/) và Database của website tại: [database](https://google.com)
   2. Các API được viết bằng ngôn ngữ PHP, nhằm cung cấp các dữ liệu được gọi từ app di động.

Hình 1: mô hình hoạt động webservice

phần 3: các chức năng của ứng dụng

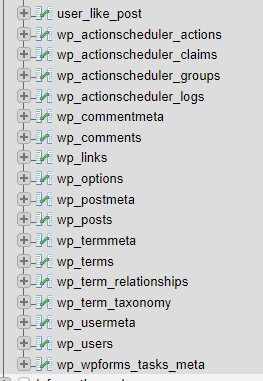
|  |  |  |
| --- | --- | --- |
| **STT** | **CHỨC NĂNG** | **MÔ TẢ** |
| 01 | Load danh mục của tất cả bài viết lên menu | Get API danh mục thừ webservice hiển thị lên thanh Drawer Navigation Menu. |
| 02 | Load tất cả biết viết, hay lọc bài viết theo từng danh mục | Get API bài viết từ webservice hiển thị lên Recycle View. Khi bấm vào từ danh mục thì các bài viết nào nằm trong danh mục đó sẽ được lọc ra. |
| 03 | Tạo tài khoản người dùng, đăng nhập, đăng xuất | Người dùng có thể tạo một tài khoản trên app bằng Email Address. Người dùng đăng nhập, đăng xuất. |
| 04 | Tìm kiếm, lọc các bài viết | Chức năng tìm kiếm các bài viết theo từ khóa. Có thể lọc từ một danh sách có sẵn hoặc tìm kiếm tất cả trên co sở dữ liệu. |
| 05 | Đa ngôn ngữ | Cho phép chọn các ngôn ngữ khác khi dùng app. |
| 06 | Xem thông tin chi tiết bài viết, like, bình luận | Khi bài viết dài, cho phép hiển hiển bài viết dưới dạng chi tiết hơn, người dùng có thể like hoặc dislike và bình luận bài viết theo ý muốn. |
| 07 | Đăng bài viết mới | Khi người dùng đăng nhập vào aap thì có chức năng đăng bài viết lên. |
| 08 | Quản lý cá nhân cho tài khoản đã đăng nhập | Người dùng có thể vào trang quản lý các nhân, quản lý tài khoản, đổi tên, mật khẩu,.. xóa các bài viết đã từng viết hoặc xóa các bình luận đã bình luận. |
| 09 | Gởi Email khôi phục tài khoản. | Khi người dùng quên mật khẩu đăng nhập. Người dùng có thể khôi phục lại tài khoản của mình bằng cách xác minh tài khoản bằng địa chỉ email từ app gởi về. |
| 10 | Thông báo bài viết mới | Khi có bài viết mới từ app sẽ gởi thông báo về. |
| 11 | Chỉnh màu theme tối hoặc sáng cho ứng dụng và sử dụng các hiệu ứng Animation | Các hiệu ứng animation được áp dụng vào cho ứng dụng, tạo sự sinh động hơn, chức năng DarkMode. |
| 12 | Tìm kiếm bài viết bằng giọng nói | Người dùng có thể tìm kiếm các bài viết bằng giọng nói. |
| 13 | Vẽ biều đồ Pie | Mô tả số lượng bài, tốc độ tăng trưởng các user đã đăng thông qua biểu đồ hình tròn. |

phần 4: bảng tổng hợp các link API dùng trong ứng dụng

1. website và database

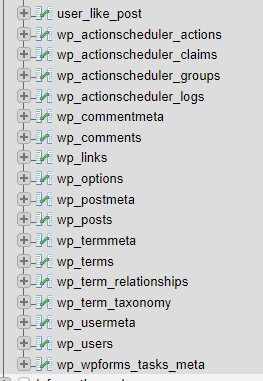
Website được tạo từ mã nguồn mở wordpress. Web có dạng blog chia sẽ các kiến thức dưới dạng đăng các bài viết.

Database của website cũng đã được wordpress tạo sẵn. Các bảng trong database như sau (Database Hình 1)



Database Hinh 1

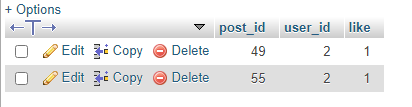
Trong đó ta chỉ cần quan tâm tới các bảng chính sau đây. Và chúng ta cũng sẽ thực hiện các thao tác chính trong các bảng này. Các bảng được gạch chân dưới hình (Database Hình 2).



Database Hinh 2

1. Bảng dữ liệu user\_like\_post

Bảng này mô tả về việc user nào like bài viết nào. Các thuộc tính trong bảng như sau(Database Hình 3).



Database Hình 3

1. Bảng dữ liệu wp\_comment

Bảng này mô tả việc các user nào comment các bài viết nào. Comment vào lúc mấy h, nội dung là gì(Database Hình 4).



Database Hình 4

1. Bảng dữ liệu wp\_post

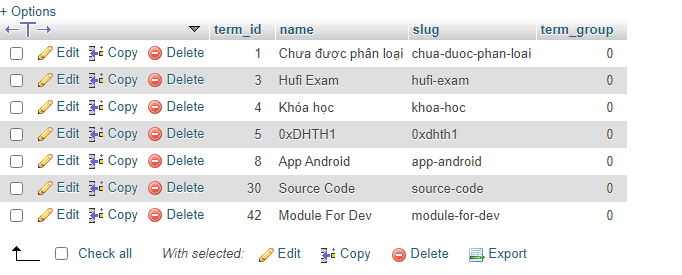
Bảng này mô tả việt các user nào đăng bài viết nào vào lúc mấy giờ, nội dung đăng gì,...(Database Hình 5)



Database Hình 5

1. Bảng dữ liệu wp\_terms

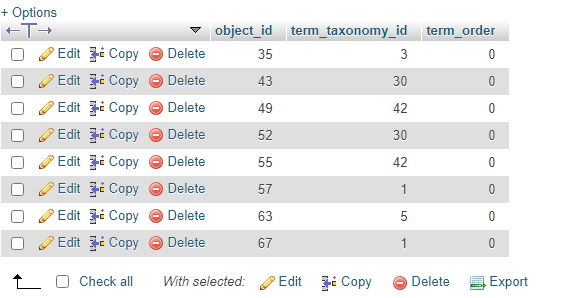
Bảng này mô tả các danh mục của bài viết. Ví dụ danh mục hufi-exam, danh mục tin học, danh mục khác,...(Database hình 6)



Database Hình 6

1. Bảng dữ liệu wp\_term\_relationships

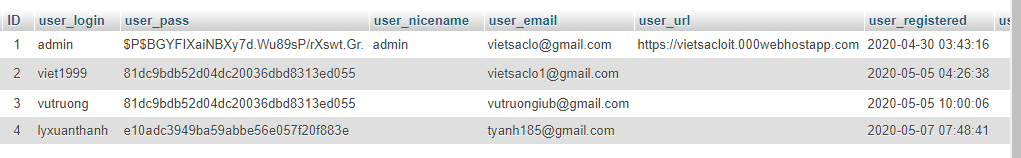
Bảng này mô tả các bài viết nào thuộc về danh mục nào. Ví dụ bài viết “các tạo thư viện trong android studio” thì bài viết này thuộc về danh mục “08DHTH1” (Database Hình 7)



Database Hình 7

1. Bảng dữ liệu wp\_users

Bảng này mô tả tất cả các user đã đăng ký tài khoản trên website (Dữ liệu được mô tả qua Database hình 8).



Database Hình 8

1. API

Các API và mô tả cho từng API (26 API)

Bảng tổng hợp 26 API

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **NAME** | **URL** | **METHOD** | **PARAMS** | **RETURN** |
| 1 | Lấy trạng thái like của user A với bài viết B | [Link](https://vietsacloit.000webhostapp.com/android_app/json_user_like_post.php) | GET | * post\_id * user\_id | * 0 chưa like * 1 đã like * -1 dislike |
| 2 | Lấy các bình luận theo bài viết | [Link](https://vietsacloit.000webhostapp.com/android_app/json_wp_comments.php) | GET | * comment\_post\_ID | * Json Object |
| 3 | Lấy bình luận theo user | [Link](https://vietsacloit.000webhostapp.com/android_app/json_wp_comments_by_user.php) | GET | * user\_id | * Json Object |
| 4 | Lấy bài viết theo ID | [Link](https://vietsacloit.000webhostapp.com/android_app/json_wp_post.php) | GET | * id | * Json Object |
| 5 | Lấy bài viết theo tiêu đề | [Link](https://vietsacloit.000webhostapp.com/android_app/json_wp_post_by_title.php) | GET | * post\_title | * Json Object |
| 6 | Lấy bài viết cho user nào đã viết | [Link](https://vietsacloit.000webhostapp.com/android_app/json_wp_post_by_user.php) | GET | * post\_author | * Json Object |
| 7 | Lấy tất cả bài viết | [Link](https://vietsacloit.000webhostapp.com/android_app/json_wp_posts.php) | GET |  | * Json Object |
| 8 | Lấy bài viết theo danh mục | [Link](https://vietsacloit.000webhostapp.com/android_app/json_wp_posts_term.php) | GET | * term\_id | * Json Object |
| 9 | Lấy tất cả danh mục | [Link](https://vietsacloit.000webhostapp.com/android_app/json_wp_terms.php) | GET |  | * Json Object |
| 10 | Lấy user theo userName hoặc Email | [Link](https://vietsacloit.000webhostapp.com/android_app/json_wp_user.php) | GET | * user\_login * user\_email | * Json Object |
| 11 | Lấy user theo ID | [Link](https://vietsacloit.000webhostapp.com/android_app/json_wp_user_by_id.php) | GET | * ID | * Json Object |
| 12 | Lấy toàn bộ user | [Link](https://vietsacloit.000webhostapp.com/android_app/json_wp_users.php) | GET |  | * Json Object |
| 13 | Đếm số lượng comment theo bài viết | [Link](https://vietsacloit.000webhostapp.com/android_app/url_count_comments.php) | GET | * comment\_post\_ID | * Json Object |
| 14 | Xóa bình luận theo id | [Link](https://vietsacloit.000webhostapp.com/android_app/url_delete_wp_comment.php) | DELETE | * comment\_ID | * 1 OK * 0 Error |
| 15 | Xóa bài viết theo ID | [Link](https://vietsacloit.000webhostapp.com/android_app/url_delete_wp_post.php) | DELETE | * ID | * 1 OK * 0 Error |
| 16 | Lấy danh mục theo id bài viết | [Link](https://vietsacloit.000webhostapp.com/android_app/url_get_term_form_wp_post.php) | GET | * ID | * Json Object |
| 18 | Thêm 1 bình luận | [Link](https://vietsacloit.000webhostapp.com/android_app/url_insert_comment.php) | POST | * comment\_post\_ID * comment\_author * comment\_author\_email * comment\_author\_url * comment\_date * comment\_content * user\_id | * 1 OK * 0 Error |
| 19 | Thêm mối quan hệ giữa bài viết và danh mục | [Link](https://vietsacloit.000webhostapp.com/android_app/url_insert_term_relationship.php) | POST | * object\_id * term\_taxonomy\_id | * 1 OK * 0 Error |
| 20 | Thêm một User | [Link](https://vietsacloit.000webhostapp.com/android_app/url_insert_user.php) | POST | * user\_login * user\_pass * user\_email * user\_registered * display\_name | * 1 OK * 0 Error |
| 21 | Thêm trạng thái user like bài viết | [Link](https://vietsacloit.000webhostapp.com/android_app/url_insert_user_like_post.php) | POST | * post\_id * user\_id * like | * 1 OK * 0 Error |
| 22 | Thêm một bài viết | [Link](https://vietsacloit.000webhostapp.com/android_app/url_insert_wp_post.php) | POST | * post\_author * post\_date * post\_modified * post\_content * post\_title * post\_name | * 1 OK * 0 Error |
| 23 | Tính tổng like theo id bài viết | [Link](https://vietsacloit.000webhostapp.com/android_app/url_sum_like_of_post.php) | GET | * post\_id | * Json Object |
| 24 | Cập nhật GUID cho bài viết | [Link](https://vietsacloit.000webhostapp.com/android_app/url_update_gui_wp_post.php) | PUT | * ID | * 1 Ok * 0 Error |
| 25 | Cập nhật trạng thái user like bài viết | [Link](https://vietsacloit.000webhostapp.com/android_app/url_update_like_post.php) | PUT | * post\_id * user\_id * like | * 1 OK * 0 Error |
| 26 | Cập nhật password cho user | [Link](https://vietsacloit.000webhostapp.com/android_app/url_update_user_password.php) | PUT | * user\_pass * user\_email | * 1 OK * 0 Error |

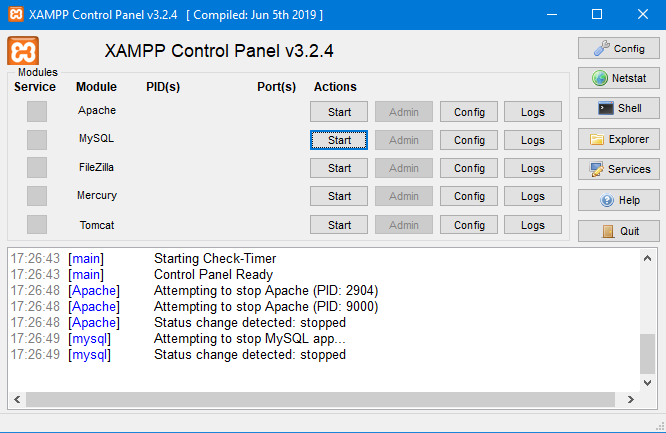
phần 5: bảng phân công đánh giá cho từng thành viên

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | MSSV | Họ Tên | Phân công | Chấm điểm | Xác nhận |
| 1 | 2001170234 | Nguyễn Quốc Việt (NT) | * Load menu, load bài viết * Tìm kiếm bằng từ khóa * Tìm kiếm bằng âm thanh * Quên mật khẩu * Xem chi tiết bài viết * Đăng ký nhanh * Animation, layout * Đăng bài viết mới | 8.5/10 | Việt |
| 2 | 2001170439 | Nguyễn Thị Ngọc Thu | * Hiển thị thông báo khi có bài viết mới * Chỉnh sửa thông tin cá nhân * Đăng nhập * Đăng ký | 8.5/10 | Thu |
| 3 | 2001170208 | Bùi Vũ Trường | * Chỉnh sửa bình luận * Vẽ biểu đồ Pie Chart * Cài đặt ngôn ngữ, cài đặt theme | 8.5/10 | Trường |

phần 6: Nội dung và các tính năng trong ứng dụng

1. các kiến thức mới cần biết
2. tạo server ảo cho web php bằng ứng dụng xampp ?

Bước 1: cài đặt phần mềm Xampp

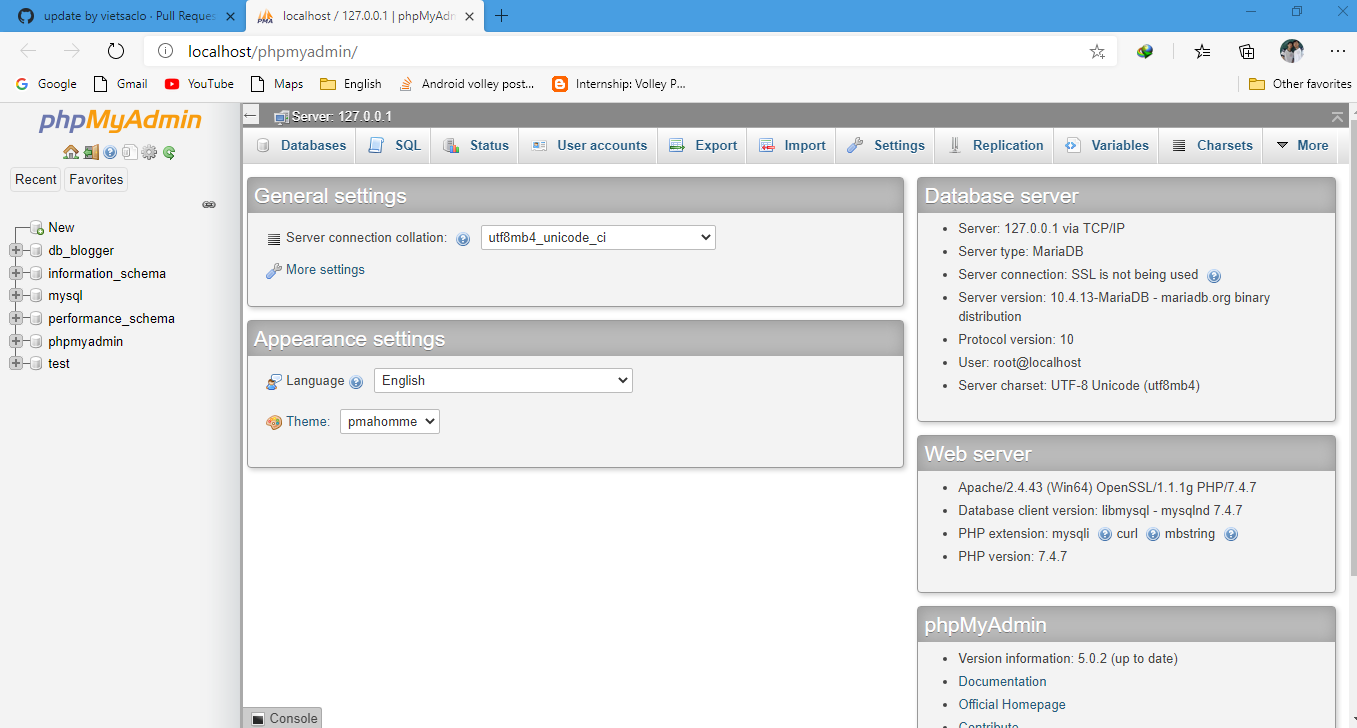


Bước 2: tại giao diện chính Xampp. Nhấn vào “start Apache” và “start MySQL”

Sau khi start Apache và MySQL, lúc này ứng dụng sẽ tạo một server ảo. Localhost:80 làm mặt định

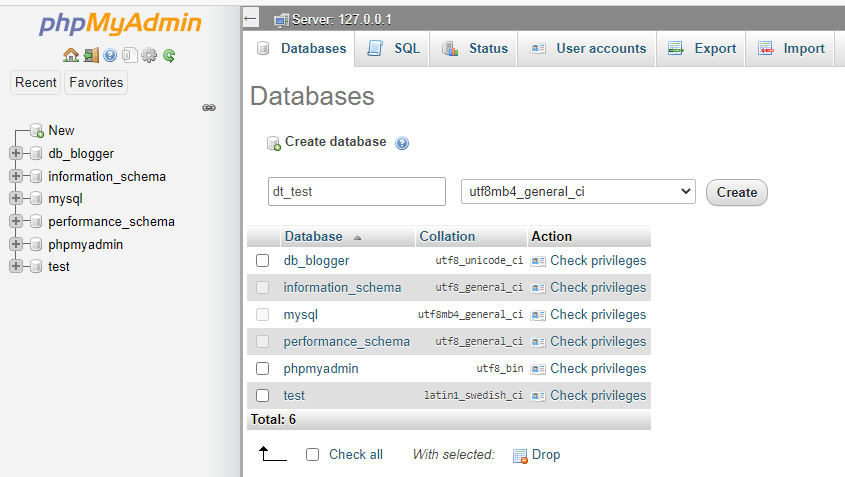
1. tạo database và tạo api ?

Bước 1: sau khi start Apache và MySql tại giao diện Xampp. Vào trình duyệt gõ vào địa chỉ: <http://localhost/phpmyadmin/>



Hình 2: giao diện phpmyadmin

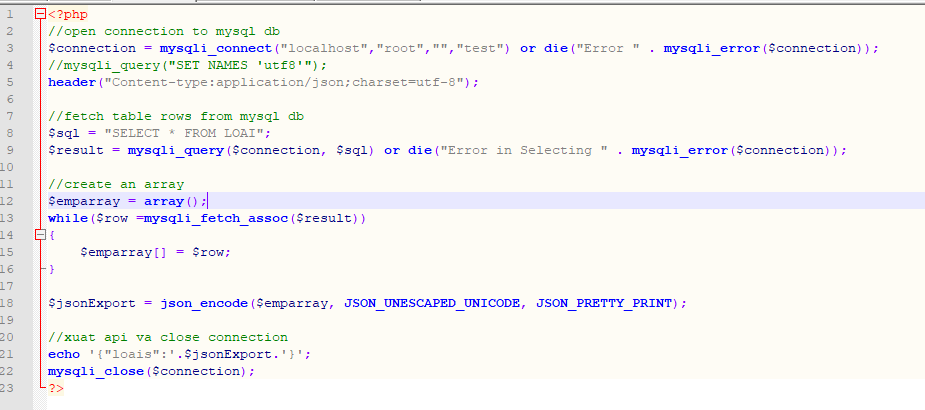
Bước 2: chọn new -> đặt tên database rồi nhấn vào nút tạo.



Hình 3: tạo database mới

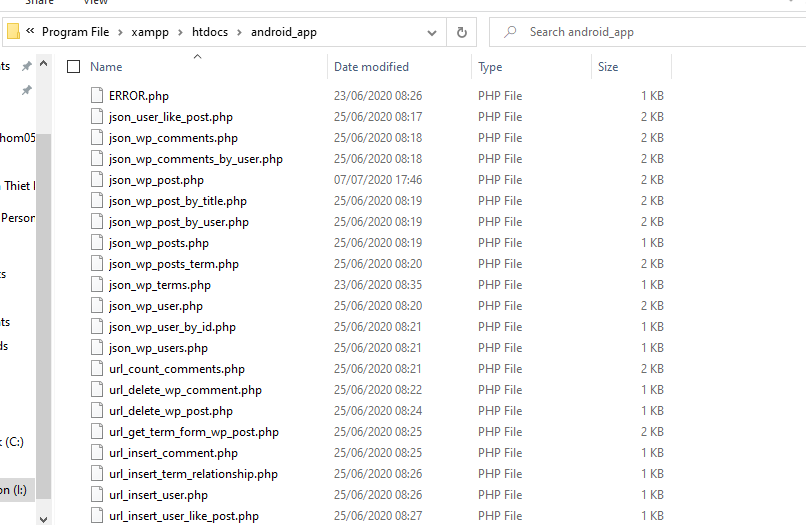
Bước 3: tạo bảng cho database: nhấn vào database cần tạo bảng và tạo các bảng cần cho ứng dụng.

Bước 4: tạo API bằng php. Cấu trúc của một file php có dạng như sau.



Hình 4: cấu trúc file API

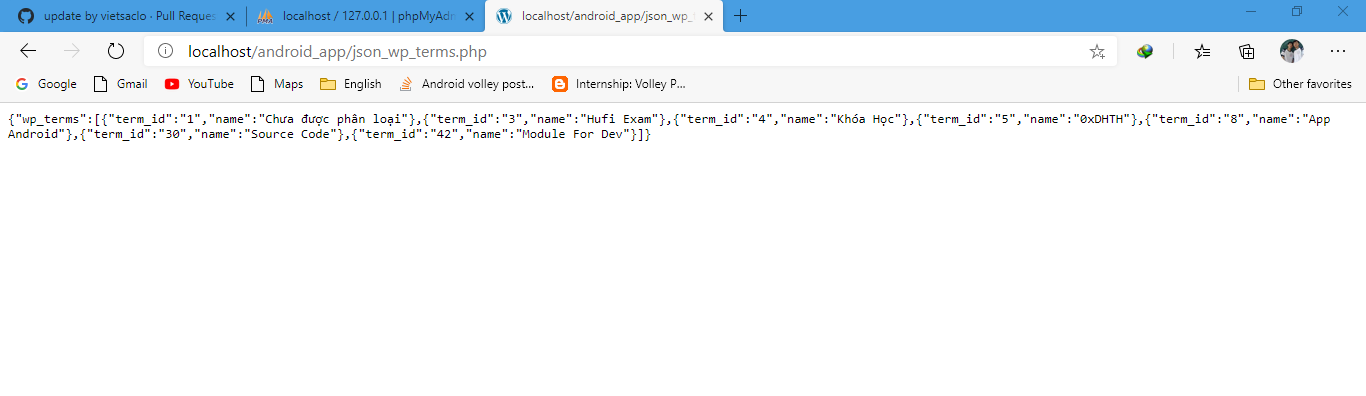
Bước 5: lưu file lại với đuôi .php => đặt file vừa lưa vào thư mục htdots ví dụ:



Hình 5: đưa file API vào thư mục htdots

1. gọi api trên mạng cục bộ ?

Bước 1: vào trình duyệt web từ localhost gõ thêm vào thên của file php API. Ví dụ: localhost/android\_app/json\_wp\_terms.php



Hình 6: gọi API từ localhost

1. tăng tốc ứng dụng bằng cách làm việc trên luồng phụ, cập nhật giao diện lên luồng chính ?

Bước 1: tạo Interface OnMyFinishListenner

**public interface** OnMyFinishListener<T> {  
 **void** onFinish(T result);  
 **void** onError(Throwable error, Object bonusOfCoder);  
}

Bước 2: tạo class TaskBackground kế thừa AsyncTask

**public abstract class** TaskBackground<T\_Result> **extends** AsyncTask<String,Integer,T\_Result> {  
 **private** OnMyFinishListener<T\_Result> **onMyFinishListener**;  
 **private** Object **bonusOfCoder**;  
  
 **public void** setBonusOfCoder(Object bonusOfCoder){  
 **this**.**bonusOfCoder** = bonusOfCoder;  
 }  
  
 **public** TaskBackground(OnMyFinishListener<T\_Result> onMyFinishListener){  
 **this**.**onMyFinishListener** = onMyFinishListener;  
 }  
  
 @Override  
 **protected void** onPreExecute() {  
 **super**.onPreExecute();  
 }  
  
 **protected abstract** T\_Result doInBackground\_Something(String... params);  
  
 @Override  
 **protected** T\_Result doInBackground(String... strings) {  
 **try**{  
 **return** doInBackground\_Something(strings);  
 }**catch** (Exception e){  
 e.printStackTrace();  
 **return null**;  
 }  
 }  
  
 @Override  
 **protected void** onPostExecute(T\_Result t\_result) {  
 **super**.onPostExecute(t\_result);  
 **if** (**onMyFinishListener** == **null**)  
 **return**;  
 **if** (t\_result == **null**)  
 **onMyFinishListener**.onError(**new** RuntimeException(**"Request Your Task Background Is Not Apply!"**), **bonusOfCoder** != **null** ? **bonusOfCoder** : KindStatusRunning.***DO\_IN\_BACKGROUND\_NOT\_RETURN\_VALUE***);  
 **else  
 onMyFinishListener**.onFinish(t\_result);  
 }  
}

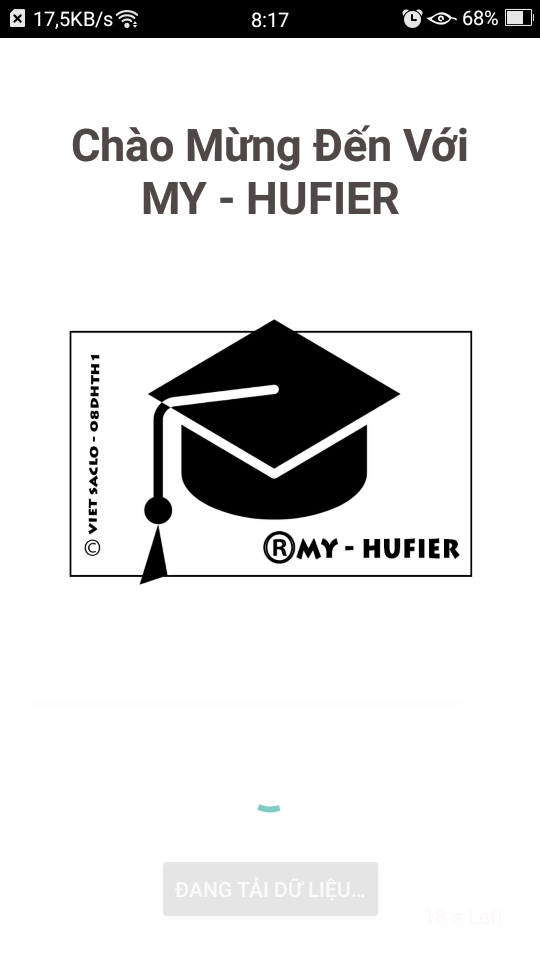
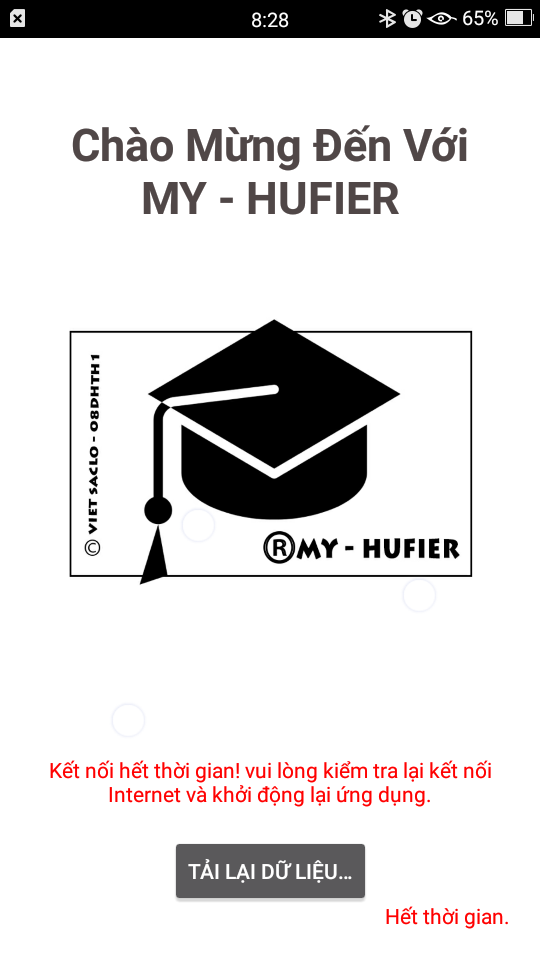
Bước 3: cài đặt trong hàm

**public static void** getStringFromInternet\_doInBackground\_AsyncTask(**final** String strUrl, **final** OnMyFinishListener<String> finish){  
 TaskBackground<String> task = **new** TaskBackground<String>(finish) {  
 @Override  
 **protected** String doInBackground\_Something(String... params) {  
 **return** *getStringFromInternet*(params[0]);  
 }  
 };task.execute(strUrl);  
}

1. Mô phỏng tất cả quy trình các activity hoạt động
2. Màn hình chờ (loading activity)

App lấy dữ liệu từ website nên phải cần dùng đến internet. Khi dùng đến internet thì việc gặp phải vấn đề loading do mất kết nối hoặc là do đường truyền trên mạng internet kém. Vì vậy cần phải có một màn hình chờ trước khi việc tải dữ liệu được hoàn tất và cũng như việc đảm bảo đường kết nối tới internet là hợp lệ.

* Giao diện màn hình chờ như sau.(Hình 8: Màn hình chờ 2)



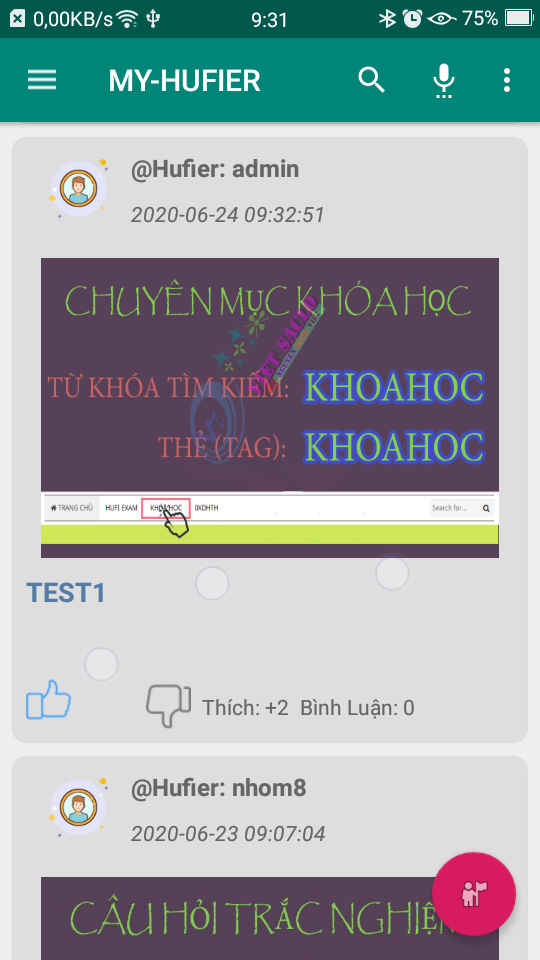
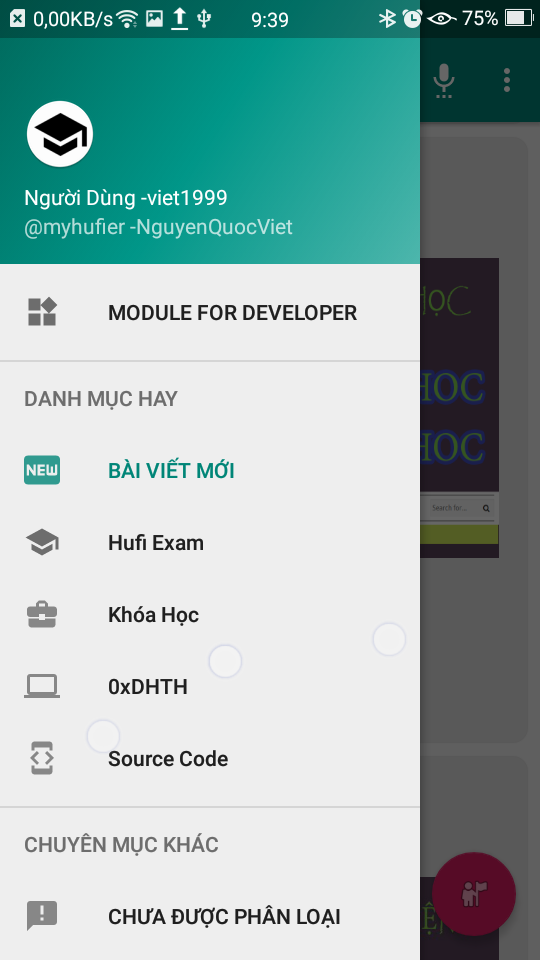
Hình 7: Màn hình chờ 2

Hình 8: Màn hình chờ 1

* Và giả sử như đến kết nối tới internet bị gián đoạn thì app sẽ không được khởi động và nhắc người dùng đảm bảo đã kết nối tới internet và tải lại app.(hình 7: Màn hình chờ 2)

1. giao diện màn hình chính (main activity)

Sau khi đã tải xong dữ liệu, sẽ hiển thị nội dung lên màn hình chính. Các nội dung gồm tải danh mục lên menu, load tất cả bài viết.



Hình 9: Tải menu

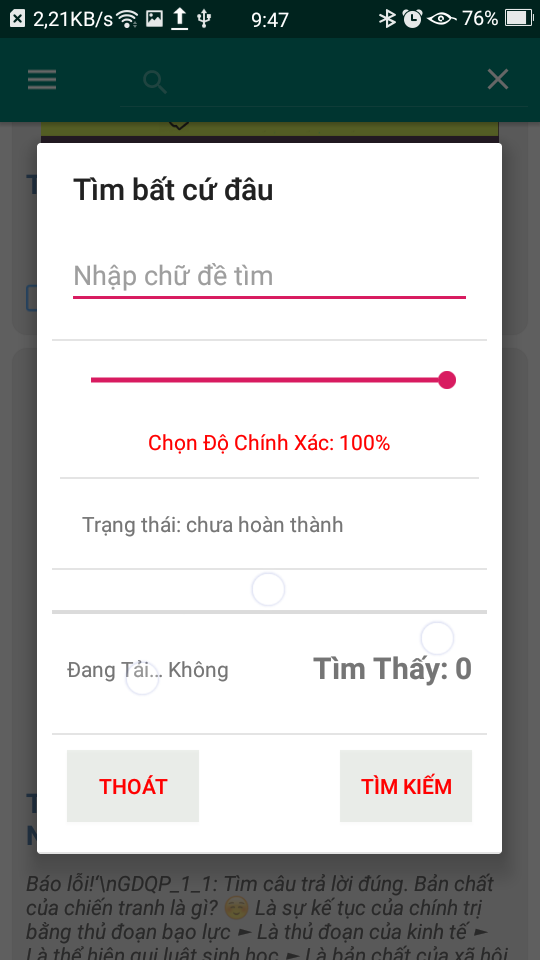
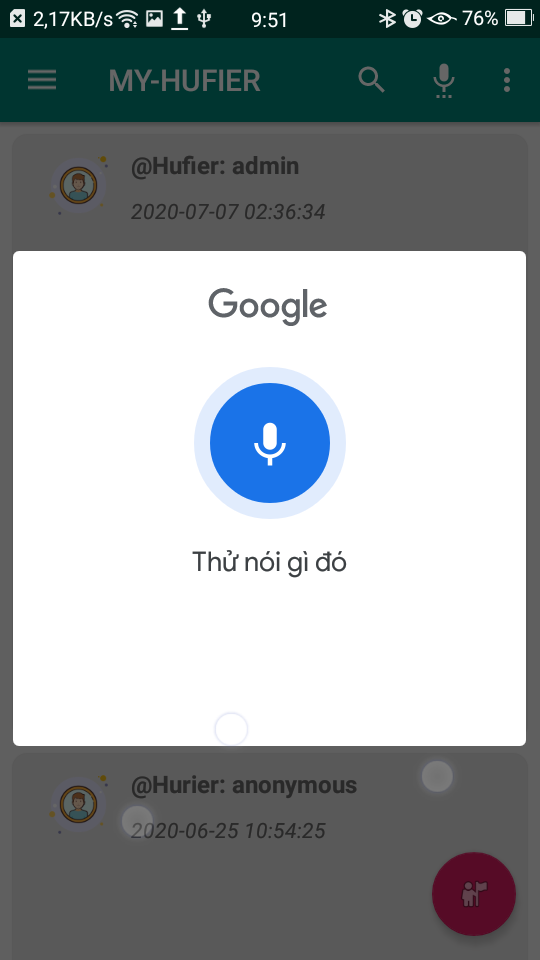
Hình 10: load bài viết

Người dùng có thể lọc các bài viết trên một danh sách có sẵn ví dụ đang chọn chuyên mục là “Hufi Exam” thì khi lọc các bài viết sẽ chỉ lấy ra từ danh mục này thôi.

Người dùng có thể tìm các bài viết trên database. Ở tính năng này người dùng có thể tìm kiếm tất cả bài viết nào với từ khóa tương tự như từ khóa để tìm.

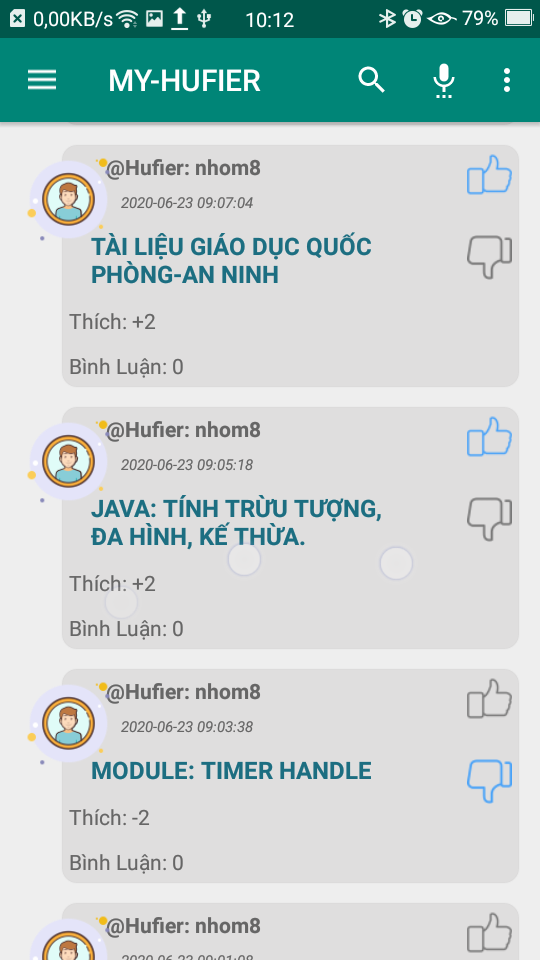
Người dùng có thể tìm các bài viết trên database bằng giọng nói (tìm kiếm bằng âm thanh).

Người dùng có thể đổi thiết kế layout tại màn hình chính với khung nhìn nhỏ gọn hơn phù hợp với việc đọc nhanh các tin mới.



Hình 12: tìm kiếm bằng từ khóa.

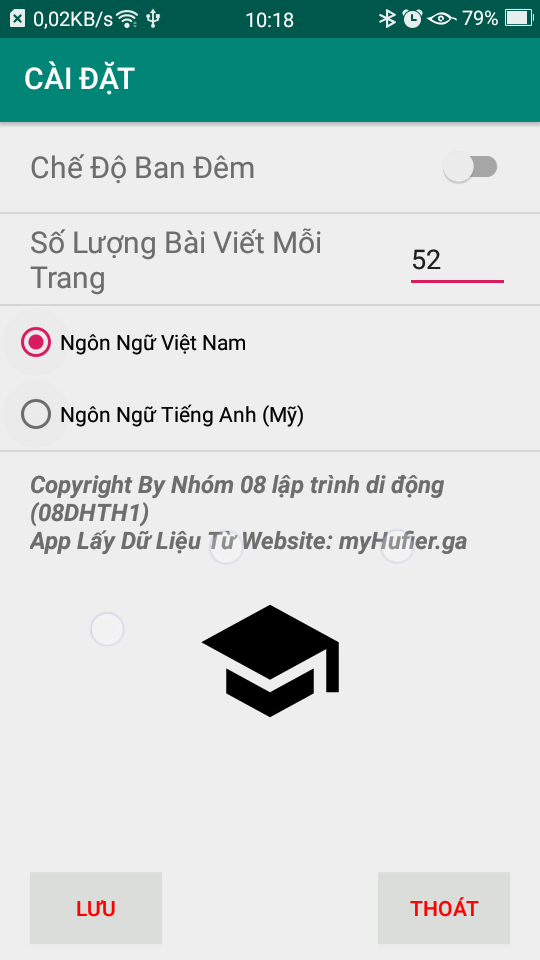
Hình 11: tìm kiếm bằng giọng nói



Hình 13: layout đọc nhanh

1. giao diện cài đặt (setting Activity)

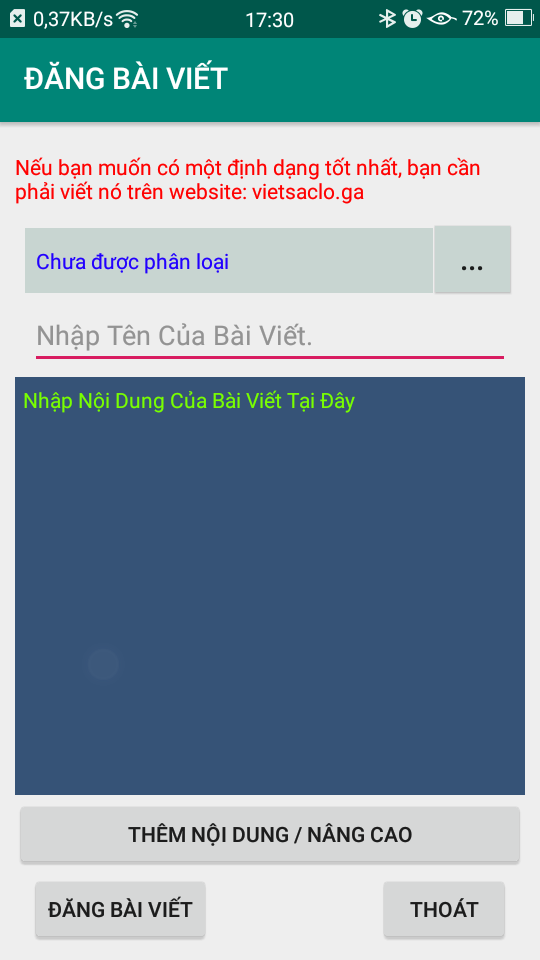
Tại đây người dùng có thể chọn theme(giao diện) sáng hoặc tối cho ứng dụng, có thể chọn số lượng bài viết cho mỗi trang, chọn loại ngôn ngữ cho ứng dụng (hỗ trợ ngôn ngữ tiếng việt và tiếng anh)



Hình 14: setting

1. giao diện đăng bài mới

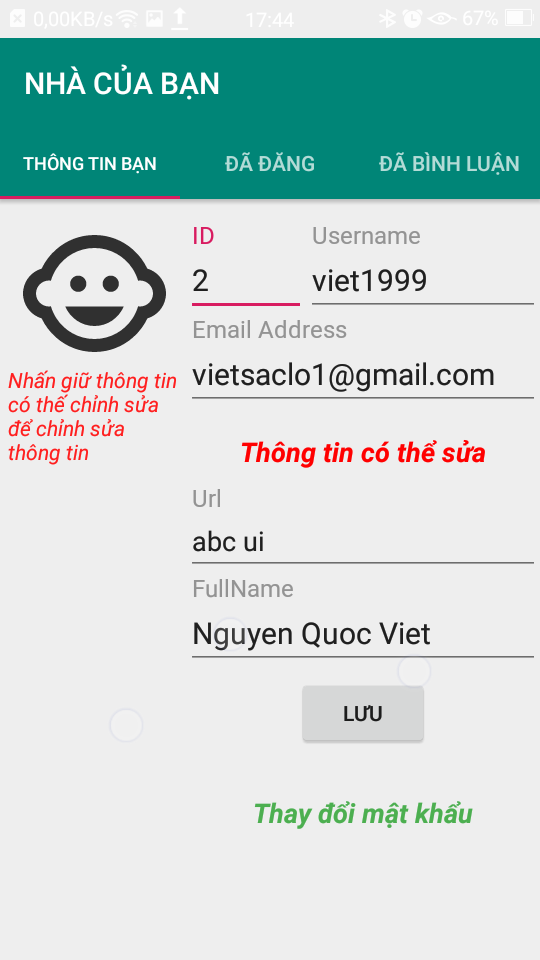
Sau khi người dùng đã đăng nhập vào trên ứng dụng. Người dùng có thể đăng bài viết mới. Bài viết này thông qua webservice sẽ được lưu trên database. Tất cả bài viết trên website và app sẽ được đồng bộ



Hình 15: đăng bài mới

1. giao diện thông tin cá nhân

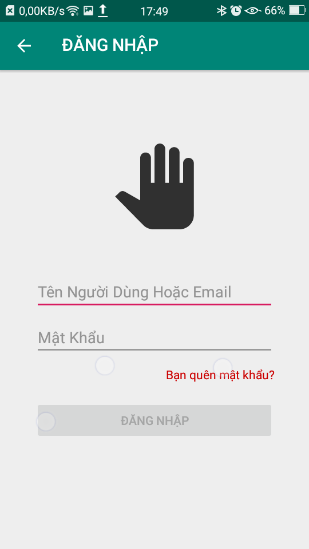
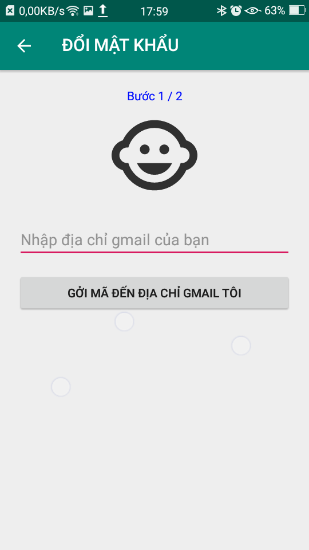
Sau khi người dùng đã đăng ký tài khoản và đăng nhập trên app. Thì người dùng có thể xem lại thông tin của mình, có thể chỉnh sửa lại thông tin như đổi tên, url cá nhân,... người dùng có thể xem lại các bài viết mình đã đăng hoặc xóa những bài viết mình đã đăng. Xem lại các bình luận mình đã bình luân, xóa các bình luận đó.



Hình 16: thông tin user

1. đăng nhập và quên mật khẩu.

Giao diện cho người dùng đăng nhập vào app. Người dùng có thể nhập tên đăng nhập hoặc địa chỉ email và mật khẩu để đăng nhập. Khi quên mật khẩu người dùng có thể khôi phục lại mật khẩu. Nhờ tính năng gởi mã khôi phục về địa chỉ email đã đăng ký.



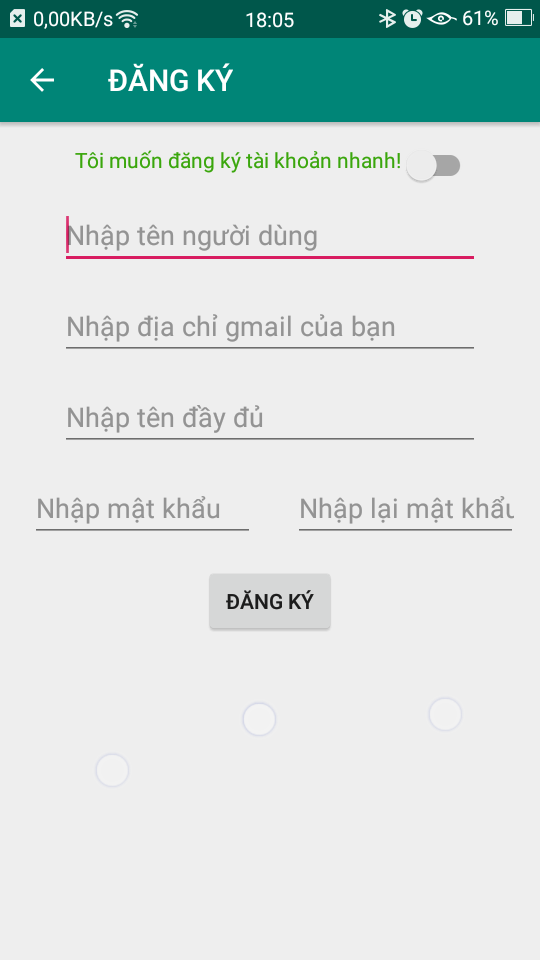
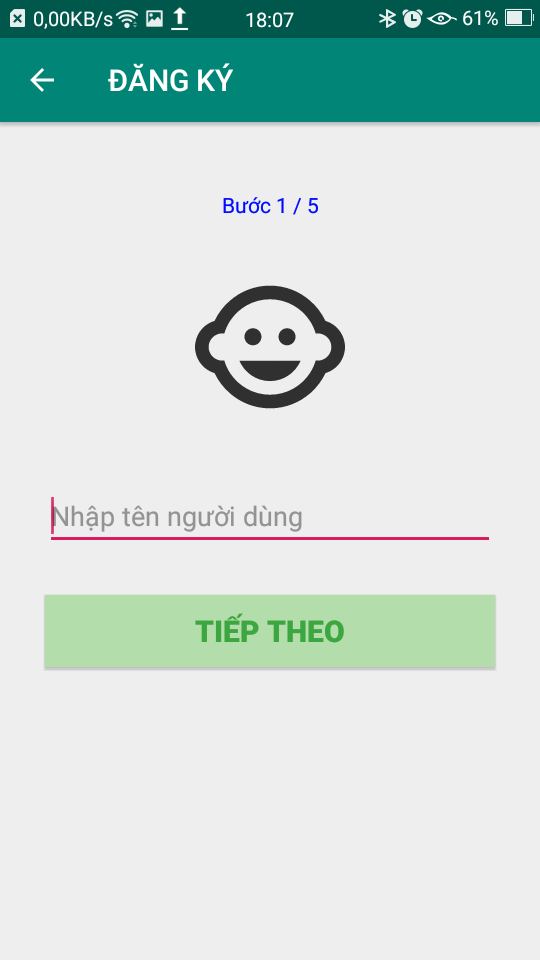
Hình 17: đổi Pass

Hình 18: đổi Pass

Hình 19: đăng nhập

1. đăng ký và đăng ký nhanh

Người dùng có thể tạo ngay một tài khoản ngay trên ứng dụng. Chọn chức năng đăng ký thường sẽ bắt buộc nhập mã bảo mật được gởi từ email. Nếu không muốn xác nhận mã từ email người dùng có thể chọn cách đăng ký nhanh.



Hình 20: đăng ký nhanh

Hình 21: đăng ký thường

1. tìm hiểu từng tính năng trong ứng dụng
2. load danh mục bài viết lên menu
3. Load tất cả biết viết, hay lọc bài viết theo từng danh mục
4. Tạo tài khoản người dùng, đăng nhập, đăng xuất
5. Tìm kiếm, lọc các bài viết
6. Đa ngôn ngữ
7. Xem thông tin chi tiết bài viết, like, bình luận
8. Đăng bài viết mới
9. Quản lý cá nhân cho tài khoản đã đăng nhập
10. Gởi Email khôi phục tài khoản
11. thông báo bài viết mới
12. Chỉnh màu theme tối hoặc sáng cho ứng dụng và sử dụng các hiệu ứng Animation
13. Tìm kiếm bài viết bằng giọng nói