Nhóm 25

Thành viên:

Đặng Minh Quang 1551010104

Đoàn Công Trực 1851010147

Phan Thiên Phú 17h1010022

Trần Võ Bảo Duy 17h1010004

Thầy: Lê Ngọc Hiếu

Chủ đề : Tạo ứng dụng tra cứu bệnh

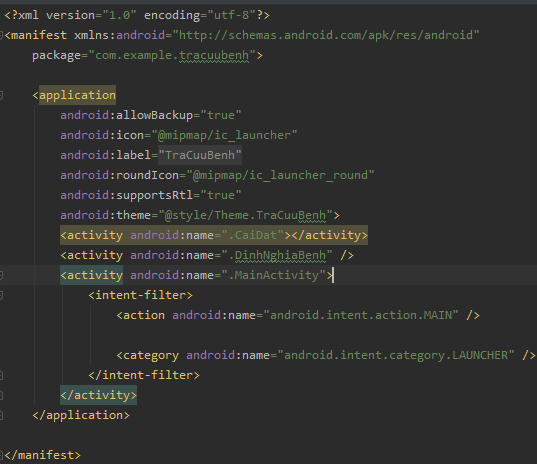
# [**quangdang142**](https://github.com/quangdang142)**/**[LTDD\_TH109-CS81\_Nhom25](https://github.com/quangdang142/LTDD_TH109-CS81_Nhom25)

**Mô tả kỹ thuật và nghiệp vụ**

Ứng dụng tra cứu bệnh được phát triển bằng ngôn ngữ Java, sử dụng IDE Android Studio.

**Các thành phần trong source** (Hoạt động của các thành phần sẽ được mô tả ở phần sau)

**1.Manifest**

****

Tập tin AndroidManifest.xml điểm danh tên các activity trong ứng dụng

**2. Java Classes**

Thư mục Fragment : bao gồm 4 fragment CachTri, PhongNgua, TrieuChung, DinhNghia. Các fragment này được hiển thị ở giao diện định nghĩa bệnh.

CaiDat.java: xử lí dữ liệu nút menu

DatabaseHelper.java: Gồm các hàm để thao tác với cơ sở dữ liệu.

DinhNghiaBenh.java: đây là 1 activity xử lí layout “activity\_dinh\_nghia\_benh.xml” (giao diện định nghĩa bệnh).

“history”: được dùng như 1 đối tượng phục vụ cho việc hiển thị lịch sử tra cứu.

LoadDatabaseAsync.java: Sử dụng TaskAsync nhằm mục đích làm mượt mà giao diện, tránh hiện tượng giật khi load cơ sở dữ liệu( hiển thị progressbar và thông báo đang tải cơ sở dữ liệu).

MainActivity.java: xử lí layout activity\_main.xml (giao diện main của app).

RecyclerViewAdapterHistory.java: xử lí lịch sử tra cứu bệnh

**3.Layout**

Gồm các layout của app

activity\_caidat.xml: giao diện khi người dùng chọn cài đặt ở menu

activity\_dinh\_nghia\_benh.xml: giao diện định nghĩa bệnh

activity\_main.xml: layout giao diện chính

alert\_dialog\_database\_copying: giao diện thông báo khi đang load database

fragment\_definition.xml: khuôn cho các fragment trong TabLayout của activity\_dinh\_nghia\_benh.xml

history\_item\_layout.xml: được hiển thị ở giao diện chính, gồm các item là các bệnh trong lịch sử tra cứu

suggestion\_row.xml: các tên bệnh có chữ cái đầu tiên trùng với các kí tự đầu tiên người dùng nhập sẽ được hiển thị ngay dưới thanh search

menu\_main.xml: giao diện cài đặt.

**4.Values**

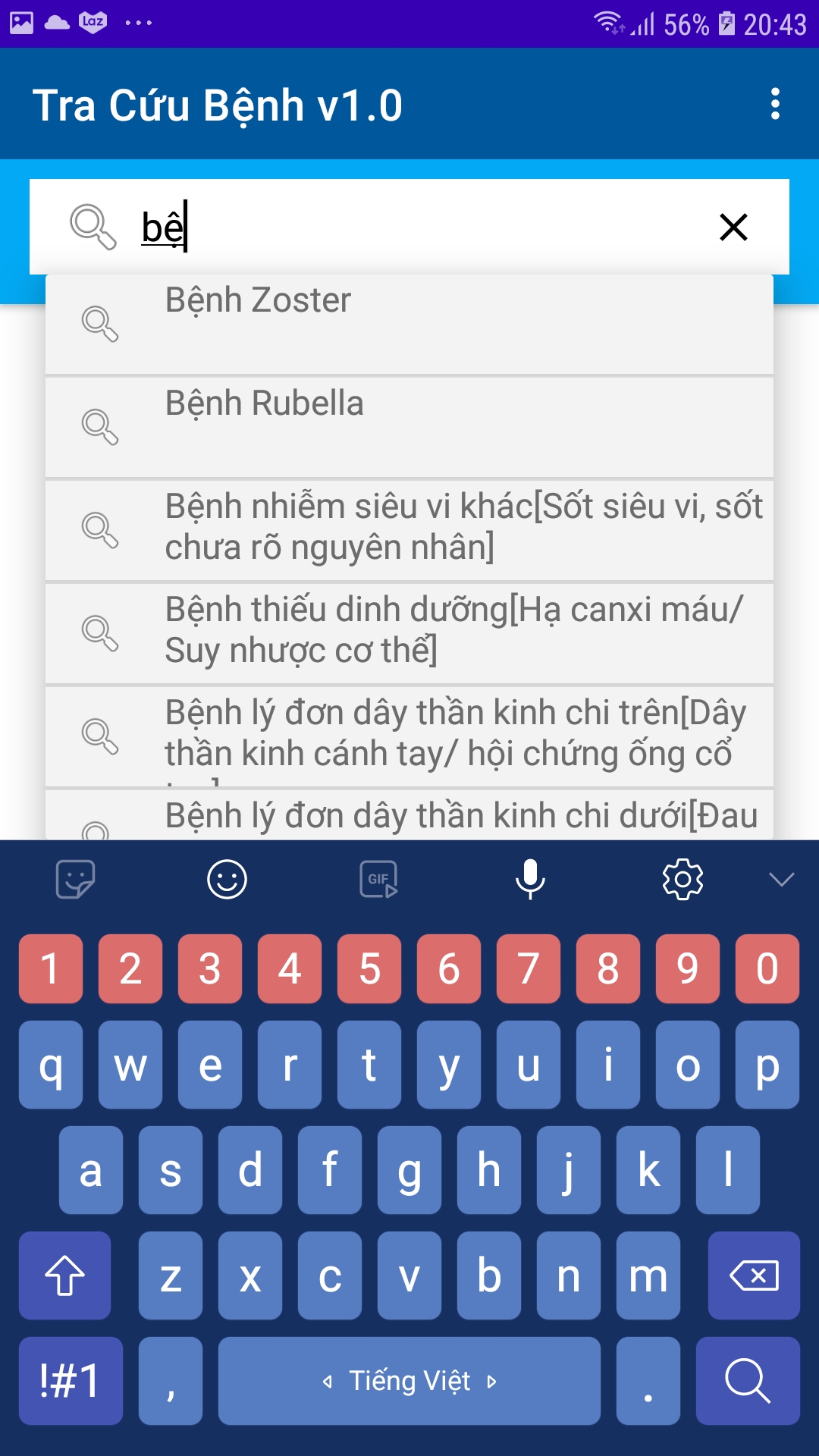
Thư mục này gồm các file xml lưu trữ định nghĩa style.

**5.Gradle Script**

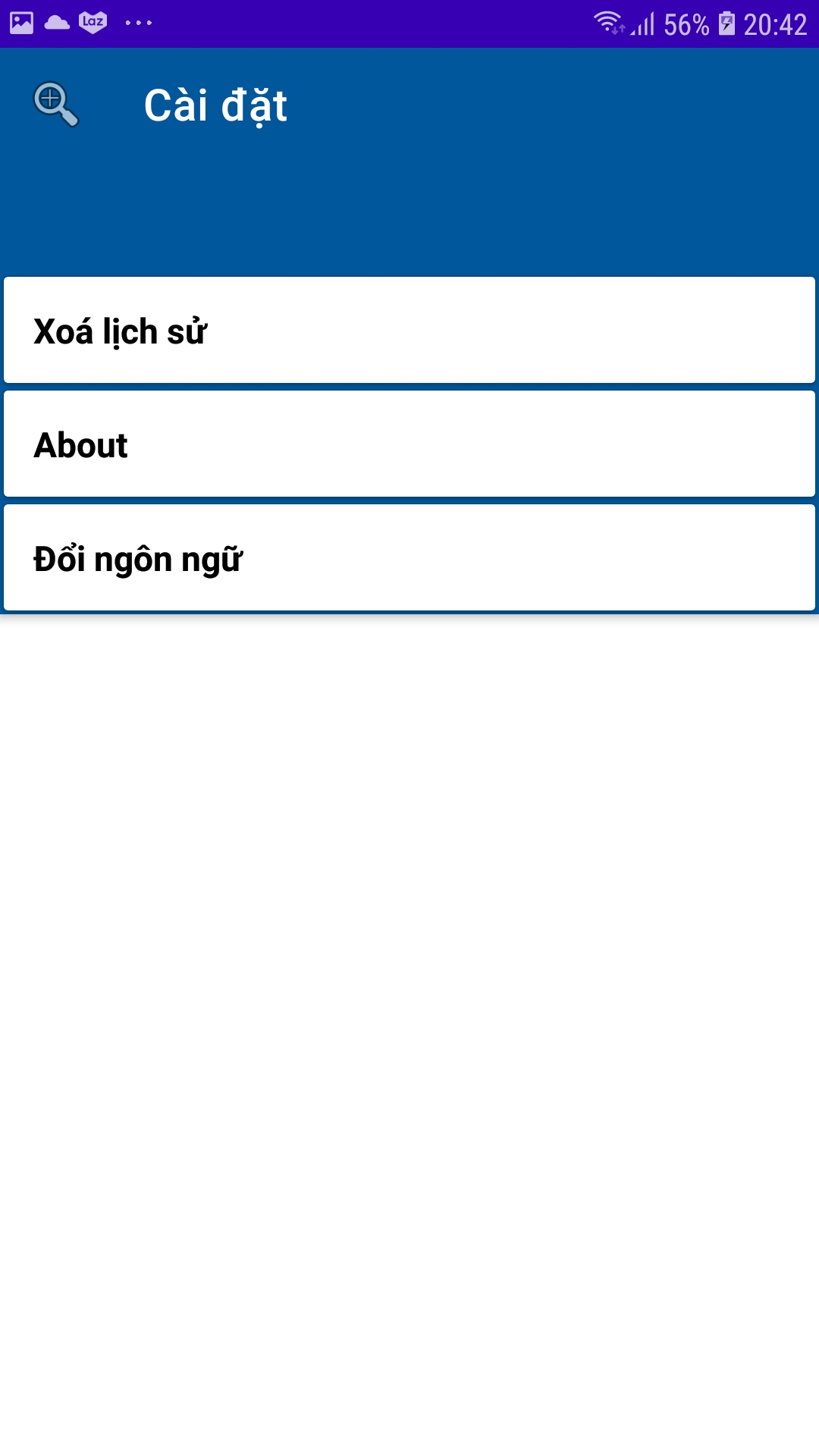
Gồm các cài đặt khi build project.

**Giao diện**

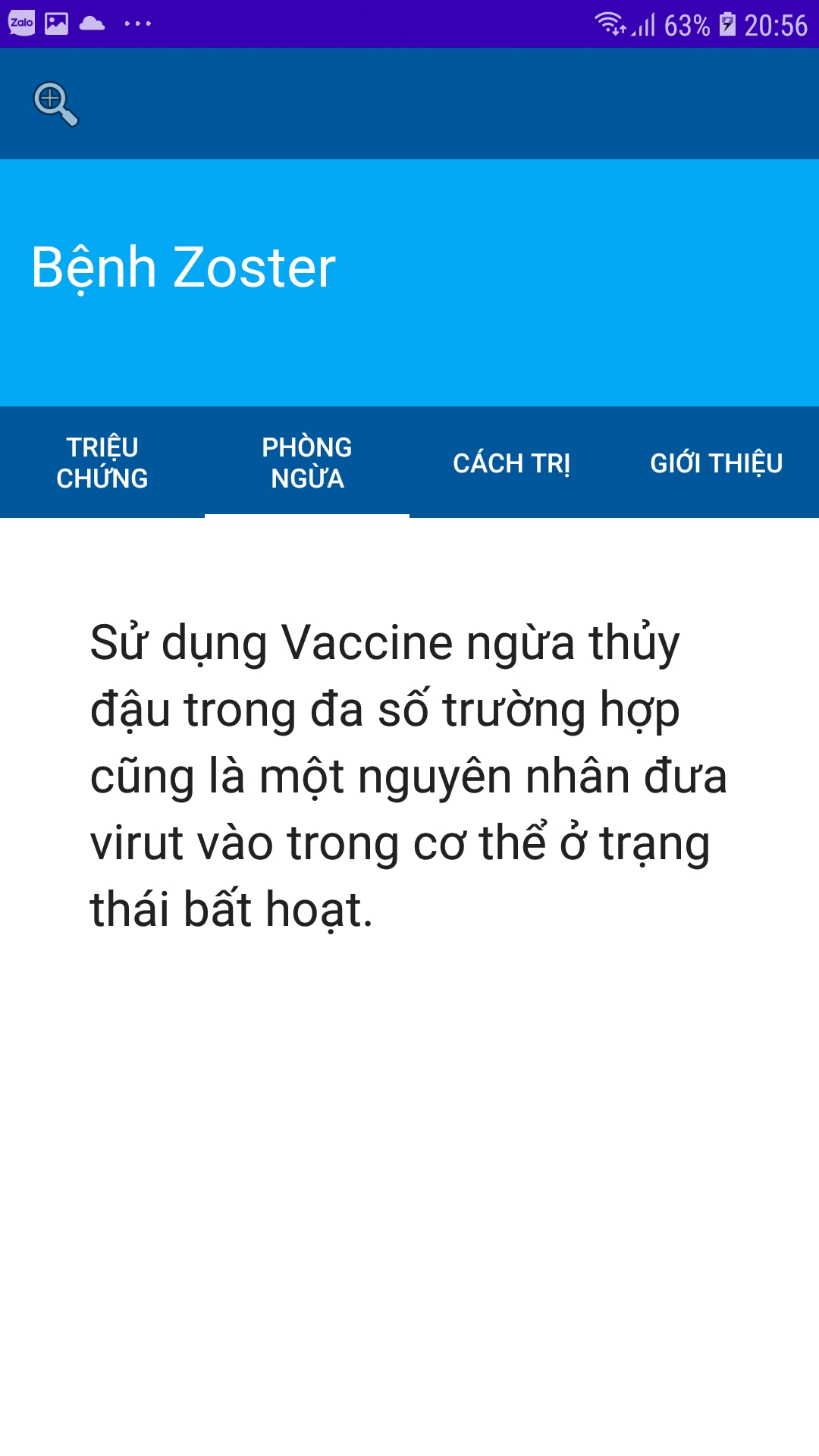
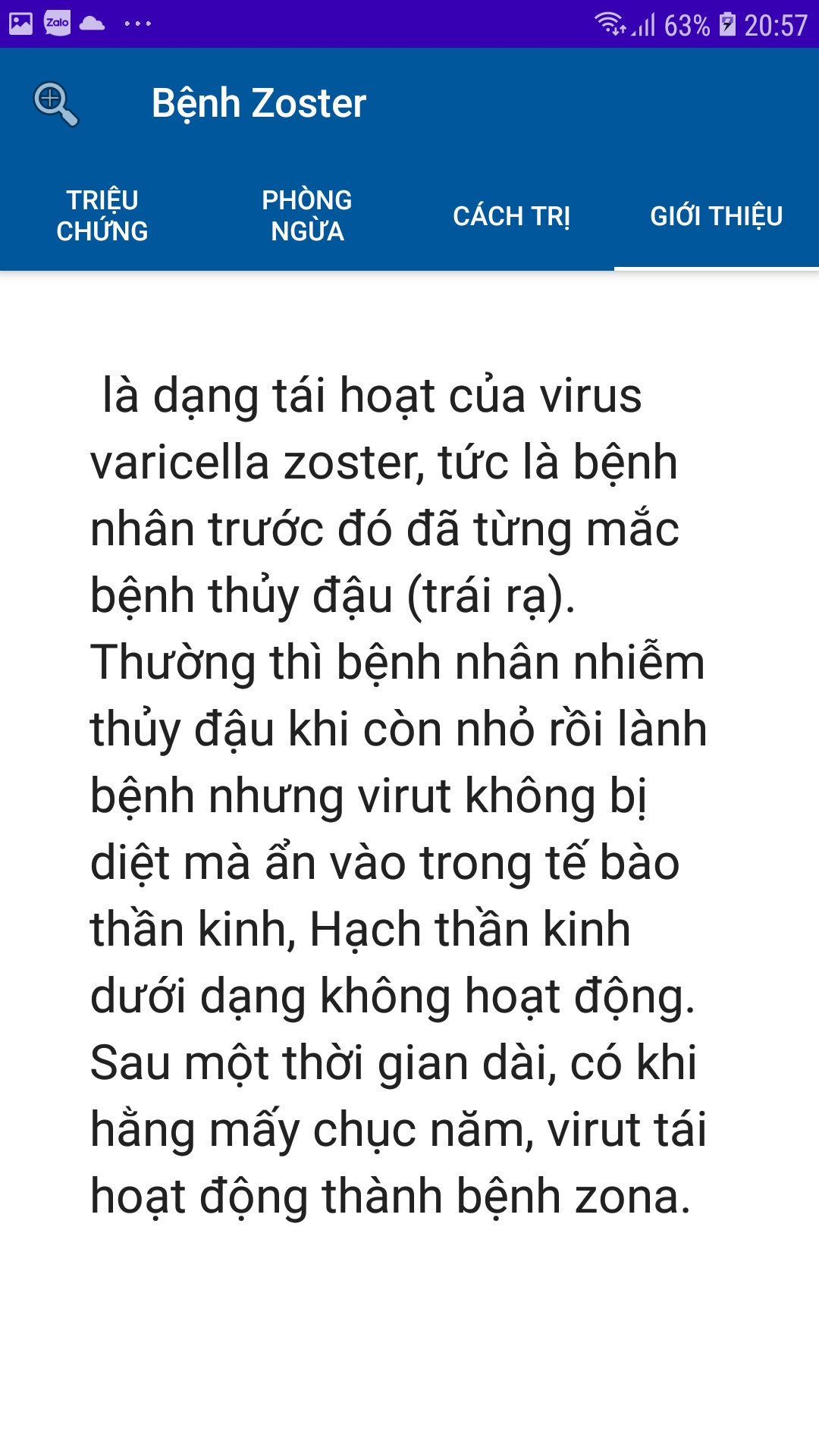
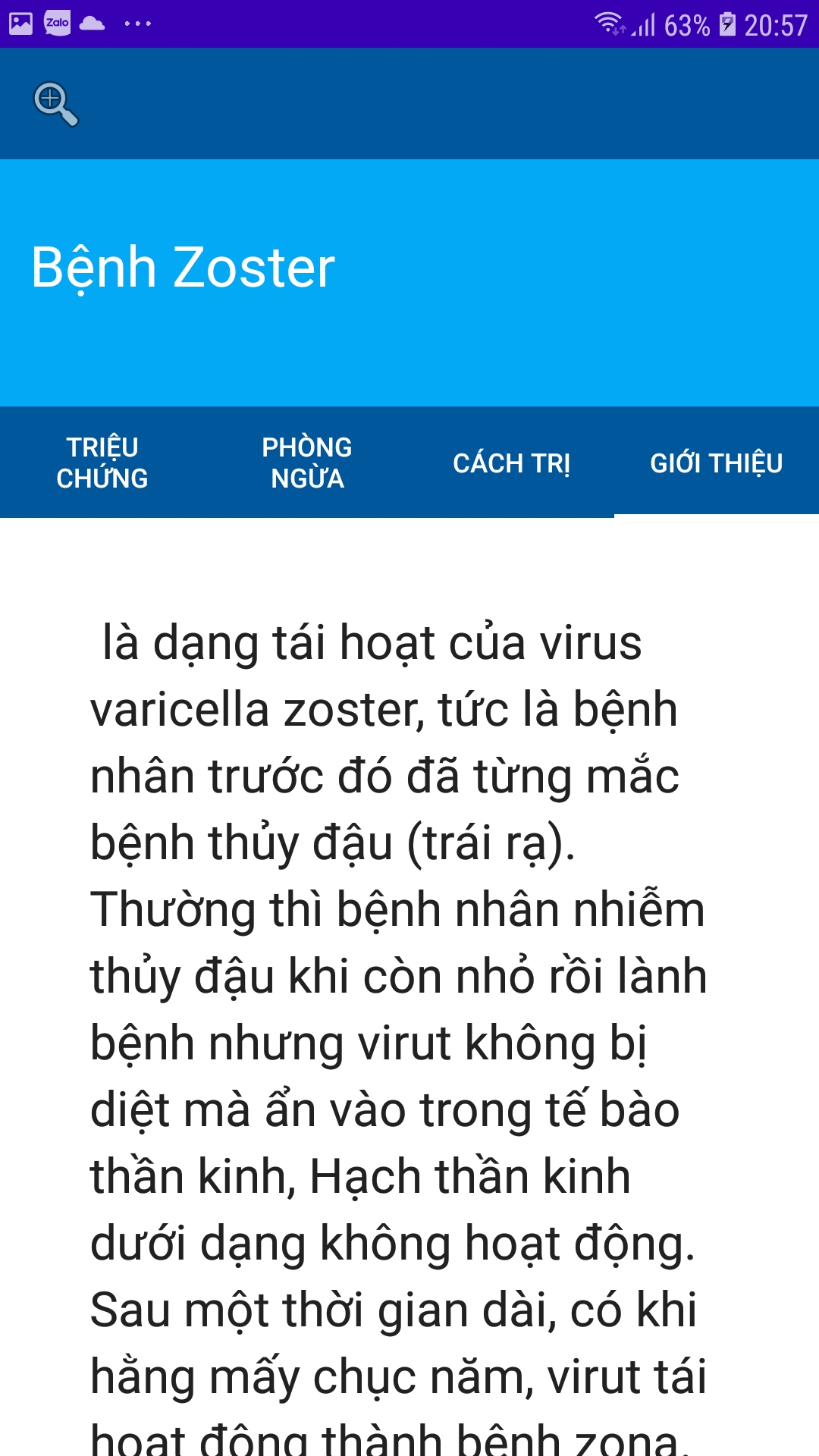
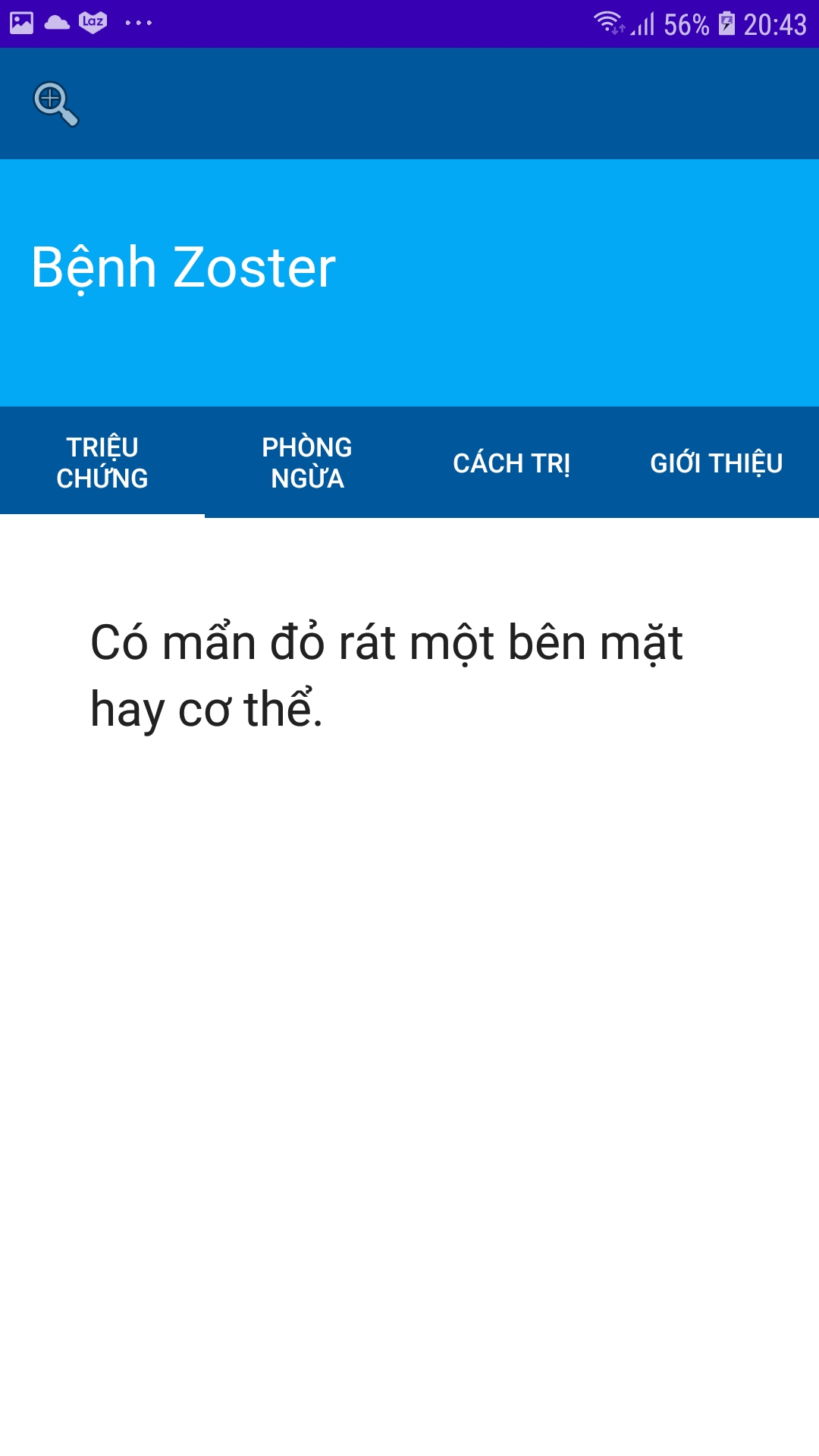
Trang chính và suggest row khi người dùng nhập tên bệnh



Menu list và giao diện cài đặt



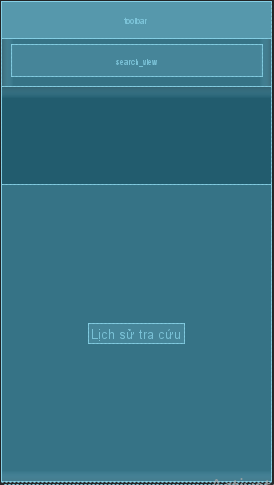
Giao diện định nghĩa bệnh



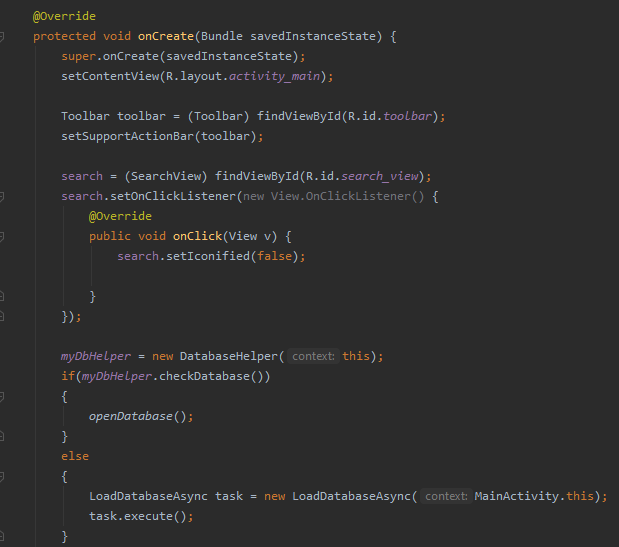
**Tương tác hoạt động giữa các thành phần trong source**

Khi người dùng mở app

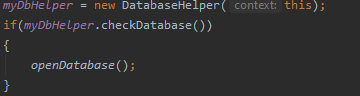
*Khung xml*

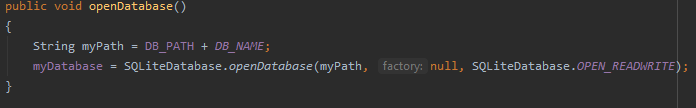


*Đoạn code chuyển xml thành java code để thao tác xử lí, gọi hàm opendatabase kết hợp với class đối tượng LoadDatabaseAsync task để kết nối với cơ sở dữ liệu.AsyncTask có tác dụng làm mượt giao diện trong quá trình tương tác với database.*

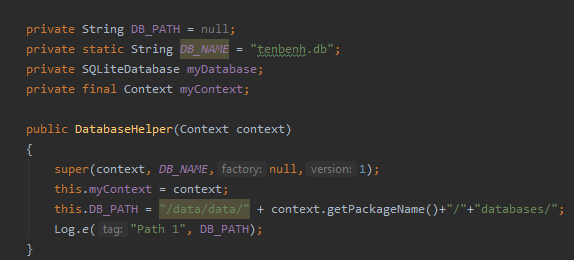


*Check database app trong bộ nhớ máy, nếu đã có*

Hàm openDatabase()



Constructor và khai báo biến,DB\_PATH là đường dẫn trong bộ nhớ máy

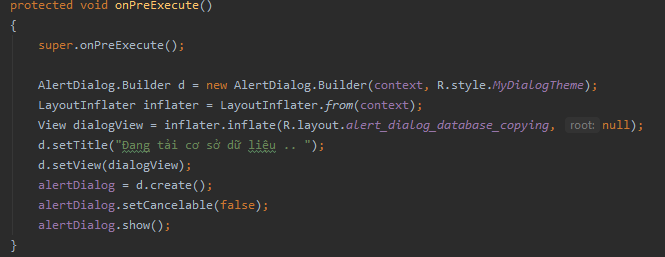


Trường hợp mới cài đặt chưa có database->copy database vào phone memory

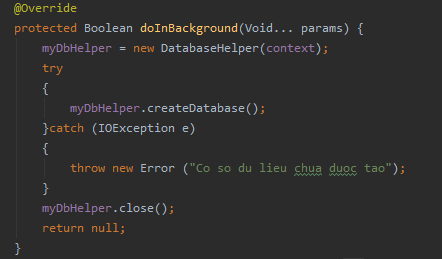


AsyncTask gồm làm 3 giai đoạn:

a.Trước khi khởi chạy, tung AlertDialog với progressbar và thông báo không thể tắt

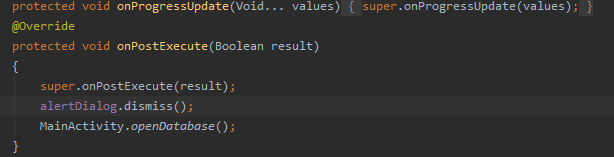


b. Chạy nền tạo đối tượng DatabaseHelper, gọi hàm createDatabase(), hàm trên sẽ gọi tiếp hàm copy để chép dữ liệu vào phone memory

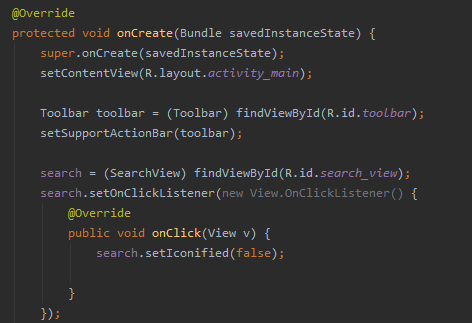


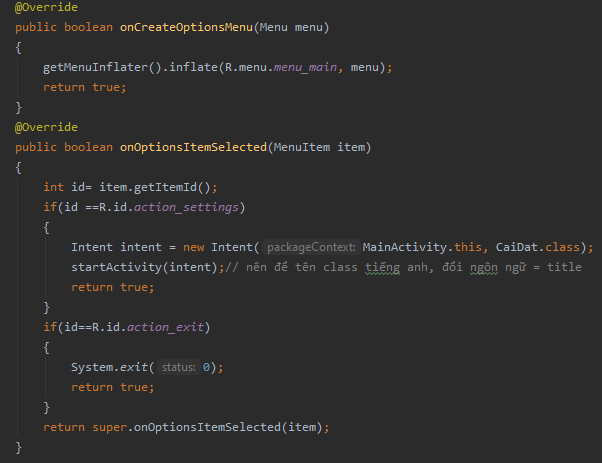


c. Sau khi chạy xong

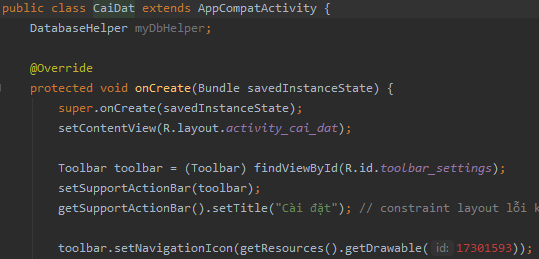


Nhận xml, gài eventlistener vào SearchView



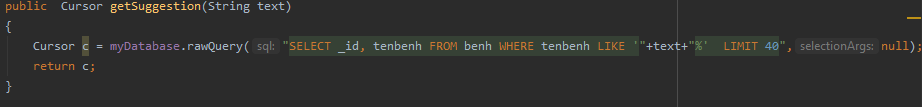
Tạo menu và xử lí khi người dùng tap menu

activity CaiDat xử lí menu (gồm cả phần xử lí lịch sử sẽ được trình bày sau)

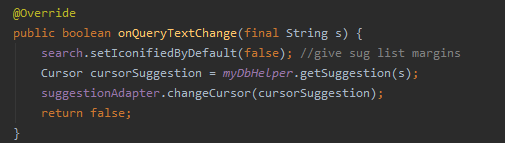


Khi người dùng nhập dùng liệu, hiện đề xuất tên bệnh như sau:

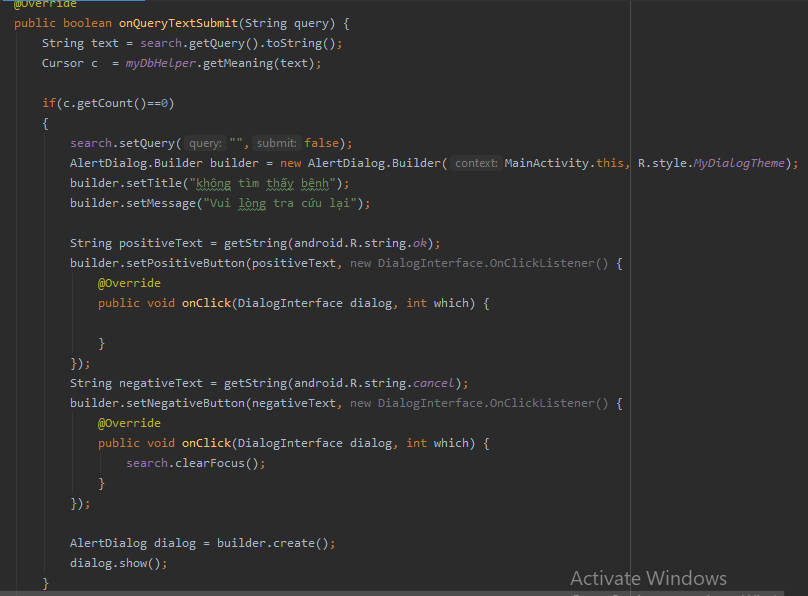
Ở DatabaseHelper



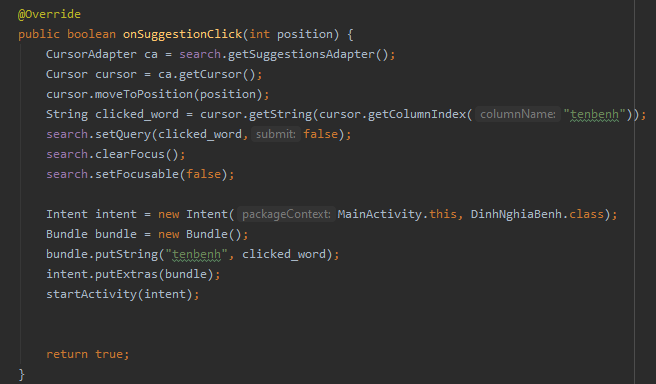
Main\_Activity



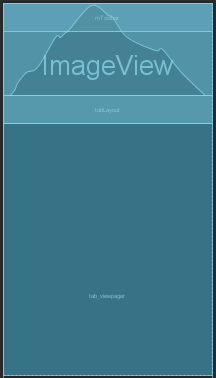
Khi người dùng submit, nếu không có tên bệnh trong database hiện alert, nếu có start DinhNghiaBenh activity



Khi người dùng chọn tên trong suggestion row cũng có thể mở DinhNghiaBenh

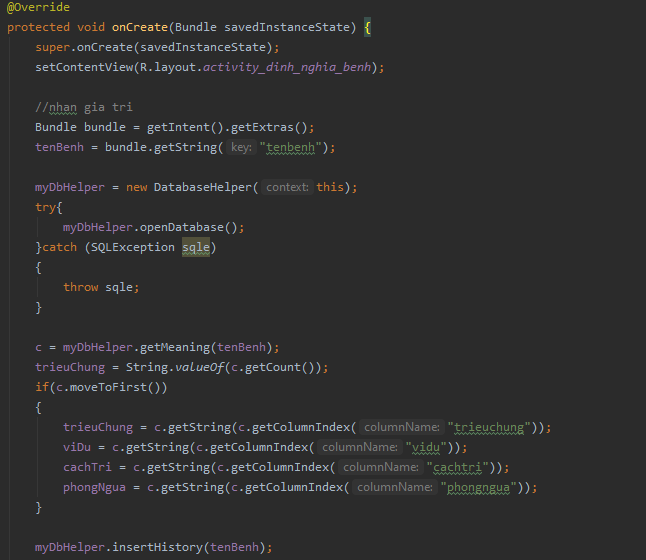


Layout xml DinhNghiaBenh



DinhNghiaBenh activity xử lí việc đẩy dữ liệu lên layout,nhét fragment vào TabLayout

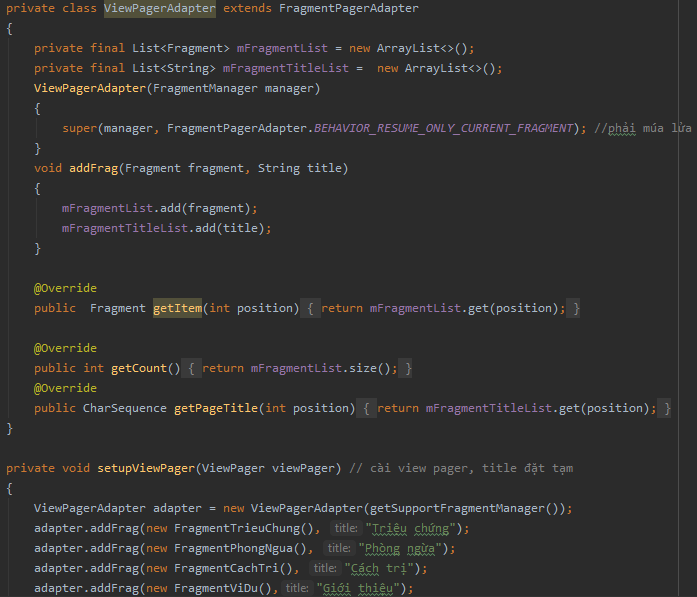
1.



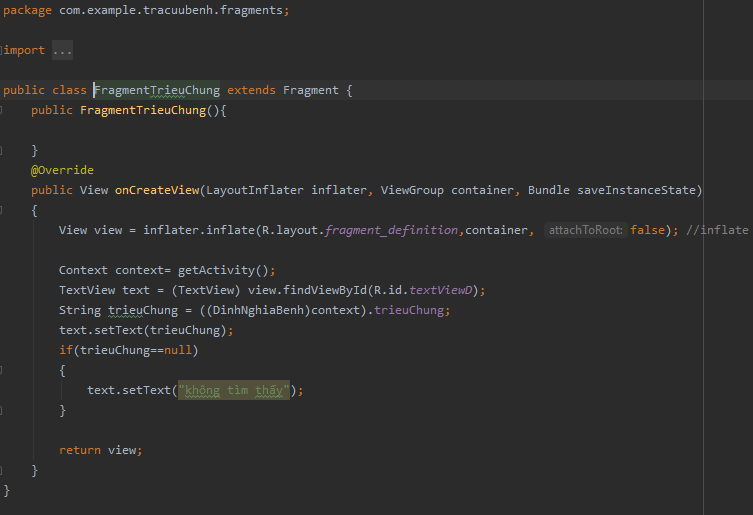
2.onTabSelected và onTabUnreleased IDE tự sinh và không được dùng



3.Hàm ViewPagerAdapter tự định nghĩa, xử lí fragment, mỗi fragment có 1 class riêng

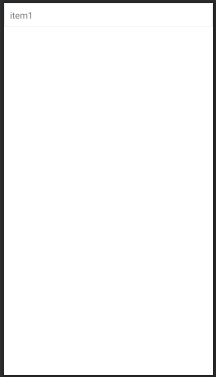


Tất cả các fragment có khuôn chung id textViewD,nhận dữ liệu từ DinhNghiaBenh và nhét vào phần của mình. Code của 4 fragments tương tự nhau, chỉ thay đổi biến



Xử lí lịch sử tra cứu.

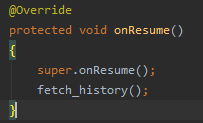
history\_item\_layout.xml,đối tượng History, lay out này nằm trong layout chính của app



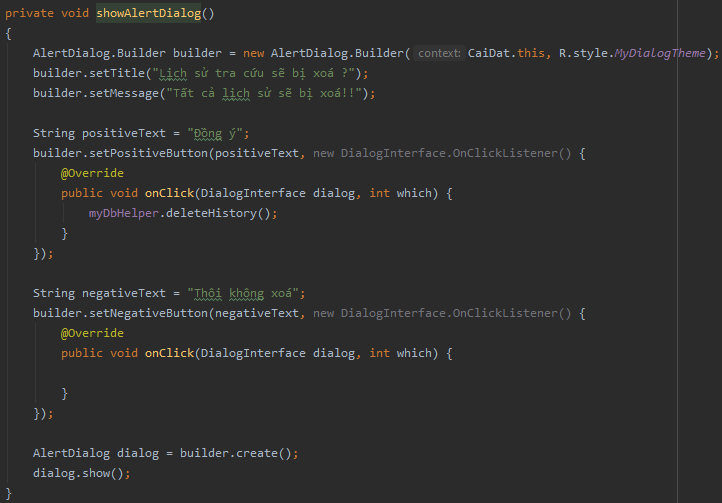
Các item được thêm vào layout bởi Main activity,dữ liệu được thêm vào bởi DinhNghiaBenh activity khi người dùng chọn tên bệnh



fetch history khi trở về giao diện chính



Xử lí xoá dữ liệu ở menu cài đặt,hàm deleteHistory của DatabaseHelper chạy querry xoá toàn bộ dữ liệu bảng history.



Hướng dẫn sử dụng

Các chức năng của app hiện tại khá đơn giản, chỉ cần nhập tên bệnh và chọn tìm kiếm. Chức năng đổi ngôn ngữ hiện tại vẫn chưa có.