**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

──────── \* ───────

**BÀI TẬP LỚN**

Môn: Phát triển ứng dụng cho thiết bị di động

**Tên đề tài**

**Ứng dụng xem bói tình yêu**

Sinh viên thực hiện: Nguyễn Thị Hải Yến – 20167045

Vương Thị Nga - 20166503

Mã lớp: 108597

Giảng viên hướng dẫn: TS. Nguyễn Hồng Quang

**Hà Nội, tháng 5 năm 2019**

Contents

[**DANH MỤC BẢNG MINH HỌA** 3](#_Toc9892113)

[**DANH MỤC HÌNH ẢNH MINH HỌA** 4](#_Toc9892114)

[**1.** **Giới thiệu đề tài** 5](#_Toc9892115)

[1.1. Mô tả đề tài 5](#_Toc9892116)

[1.1.1. Mục tiêu 5](#_Toc9892117)

[1.1.2. Đối tượng sử dụng 6](#_Toc9892118)

[1.2. Các kỹ thuật Android đã áp dụng 6](#_Toc9892119)

[**2.** **Phân tích thiết kế** 8](#_Toc9892120)

[2.1. Biểu đồ Use case 8](#_Toc9892121)

[2.2. Biểu đồ hoạt động 9](#_Toc9892122)

[2.3. Biểu đồ trình tự 10](#_Toc9892123)

[**3.** **Các chức năng** 11](#_Toc9892124)

[3.1. Chức năng Hướng dẫn 11](#_Toc9892125)

[a. Code và giải thích 11](#_Toc9892126)

[b. Giao diện minh họa 13](#_Toc9892127)

[3.2. Chức năng Bói theo tên 13](#_Toc9892128)

[a. Code và giải thích 13](#_Toc9892129)

[b. Giao diện minh họa 24](#_Toc9892130)

[3.3. Chức năng Bói theo năm sinh 25](#_Toc9892131)

[a. Code và giải thích 25](#_Toc9892132)

[b. Giao diện minh họa 30](#_Toc9892133)

# **DANH MỤC BẢNG MINH HỌA**

[Bảng 1: Đặc tả use case Hướng dẫn 8](#_Toc9892011)

[Bảng 2: Đặc tả Use case Bói theo tên 9](#_Toc9892012)

[Bảng 3: Đặc tả Use case bói theo năm sinh 9](#_Toc9892013)

# **DANH MỤC HÌNH ẢNH MINH HỌA**

[Hình 1: TabHost 7](#_Toc9892046)

[Hình 2: Biểu đồ Use case của ứng dụng 8](#_Toc9892047)

[Hình 3: Biểu đồ hoạt động của ứng dụng 10](#_Toc9892048)

[Hình 4: Biểu đồ trình tự của ứng dụng 10](#_Toc9892049)

[Hình 5: Giao diện Hướng dẫn 13](#_Toc9892050)

[Hình 6: Giao diện bói theo tên 25](#_Toc9892051)

[Hình 7: Giao diện bói theo Năm sinh 31](#_Toc9892052)

# **Giới thiệu đề tài**

## Mô tả đề tài

### Mục tiêu

Ngày mà công nghệ chưa phát triển, có một trò chơi giải trí sau những giờ học căng thẳng mà hầu như thế hệ 8x, 9x nào cũng biết tới và chơi. Đó chính là trò bói tình yêu bằng cách đếm chữ cái trong tên và cộng lại, chia tỉ lệ phần trăm giữa tên của hai người. Nếu tỉ lệ phần trăm cao tương đương hai người được bói có duyên với nhau, ngược lại, nếu tỉ lệ phần trăm thấp thì có nghĩa hai người không có duyên với nhau. Và đó cũng chỉ là trò giải trí, giải tỏa áp lực học hành của học sinh, cũng là tăng thêm sự gắn kết tình bạn giữa các thành viên trong lớp. Tuy nhiên, việc viết trên giấy và thực hiện cộng, trừ, nhân, chia gây mất nhiều thời gian cũng như lãng phí nguồn giấy, mực của học sinh. Nhận thấy trò chơi một thời này đã trở thành bất hủ và chưa bao giờ hết nóng trong giới học sinh, sinh viên. Với sự phát triển nhanh chóng mặt của công nghệ và kiến thức được học trong môn “Phát triển ứng dụng cho thiết bị di động”, nhóm em đã xây dựng ứng dụng “Bói tình yêu”. Ứng dụng này vẫn giữ được nguyên vẹn logic của trò chơi, nhưng nó giúp cho trò chơi trở lên thuận tiện hơn, không lãng phí giấy, mực của người chơi, hơn cả là giúp người chơi không phải tính toán thủ công rắc rối, mất thời gian. Để tăng thêm tính đa dạng, phong phú cho người chơi, nhóm em phát triển thêm tính năng xem bói qua năm sinh, giúp trò chơi không gây nhàm chán với chỉ một hình thức bói. Dữ liệu bói được tổng hợp từ các nguồn tin cậy, chính xác, giúp tăng thêm sự tin cậy từ người dùng với ứng dụng.

Chức năng chính của ứng dụng gồm:

* Hướng dẫn chơi trò chơi
* Bói theo tên
* Bói theo năm sinh

Để chơi được trò chơi một cách chuẩn xác, ứng dụng có cung cấp một tab hướng dẫn cách chơi. Từ đó người chơi có thể lựa chọn “bói theo tên” hoặc “bói theo năm sinh” theo các tab tương ứng trên ứng dụng. Để xem kết quả người chơi cần nhập đầu vào là tên của họ và người ấy hoặc năm sinh của họ và người ấy. Ứng dụng sẽ tính toán và đưa ra tỉ lệ kết quả và những dòng thông báo tương ứng với kết quả đó.

Ứng dụng có giao diện tương đối bắt mắt, màu sắc hài hòa, điều hướng rõ ràng giúp người dùng thoải mái nhất khi dùng. Ứng dụng cũng không yêu cầu người dùng đăng nhập, tạo tài khoản, khai báo thông tin một cách mất thời gian mà người dùng sử dụng ứng dụng có thể thực hiện chức năng chính của hệ thống. Mục tiêu của ứng dụng là lấy được sự hài lòng nhất của người dùng về ứng dụng.

Dung lượng ứng dụng khá nhẹ, dẫn đến việc cài đặt một cách nhanh chóng và dễ dàng, giúp người dùng tiết kiệm được dữ liệu một cách tối đa. Từ đó thu hút người dùng đến với ứng dụng.

### Đối tượng sử dụng

Đối tượng sử dụng chính mà ứng dụng hướng tới:

* Học sinh (Đối tượng chính)
* Sinh viên (Đối tượng chính)
* Đối tượng khác (Nhóm nhân viên làm thêm partime, nhân viên văn phòng, …)

Trò chơi có nguồn gốc từ học sinh, sinh viên ngày xưa nên đối tượng chính của ứng dụng chính là những sinh viên, học sinh, những người biết đến trò chơi và hiểu rõ trò chơi. Tiếp cận những đối tượng học sinh, sinh viên rất dễ dàng, và môi trường của họ cũng là thị trường rộng để sử dụng ứng dụng. Tuy nhiên, những đối tượng như nhân viên làm parttime, nhân viên văn phòng cũng là đối tượng không nằm ngoài phạm vi sử dụng ứng dụng. Bởi vì môi trường làm việc của họ cũng để ứng dụng có chỗ đứng. Cụ thể, nhân viên parttime cũng có những thời gian rảnh để cùng những bạn bè khác trong cùng nơi làm chơi trò chơi. Độ tuổi của họ thường là trẻ. Vì vậy, ứng dụng có thể là nơi họ giải trí với nhau. Đối với nhân viên văn phòng, tuy độ tuổi không còn trẻ, nhưng quá khứ, trò chơi này cũng gây ấn tượng sâu sắc với họ, từ đó, sau nhữn giờ làm việc căng thẳng, nó cũng có thể là trò chơi giải trí góp phần không nhỏ gây những tiếng cười, xóa tan mệt mỏi. Đặc biệt là đối với các “cô nàng” và “anh chàng” cô đơn.

## Các kỹ thuật Android đã áp dụng

Layout:

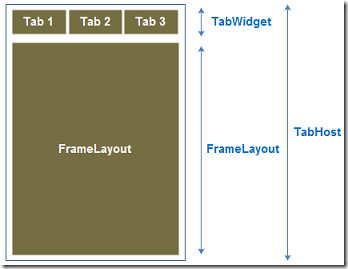
- LinearLayout: LinearLayout là một layout mà nó sắp xếp các view con một cách liên tục theo chiều ngang hoặc đứng, có thể điều chỉnh kích thước view con theo giá trị trong số layout weight.

- TextView

- EditText

- Button

- Tabhost Layout:Một TabHost Layout gồm có 3 phần: TabHost, TabWidget và FrameLayout



Hình 1: TabHost

* TabHost: Là container chính chứa các Tab buttons và Tab contents
* TabWidget: Để định dạng cho các Tab buttons : Nhãn , icon, …
* FrameLayout: là container để chứa các layout cho Tab contens.

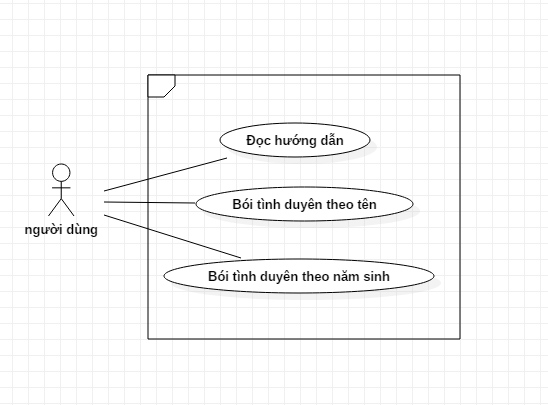
Trong phạm vi bài tập lớn, với những kiến thức đã học chúng em đã sử dụng kết hợp những lại layout khác nhau để tạo ra giao diện đẹp mắt và giúp người dùng dễ dàng sử dụng. Trong đó loại layout được sử dụng nhiều nhất là LinearLayout do nó dễ dàng sử dụng mà có hiệu quả tốt trong việc căn chỉnh các widget element.

**Các kết quả đạt được**: Ứng dụng gồm ba chức năng chính tương ứng với 3 tabhost đó là:

* Hướng dẫn chơi: Giúp người dùng hiểu rõ luật chơi.
* bói tình duyên theo tên: Người chơi nhập tên của hai người và xem kết quả.
* bói tình duyên theo năm sinh: Người chơi nhập năm sinh của hai người và xem kết quả.

# **Phân tích thiết kế**

* 1. Biểu đồ Use case



Hình 2: Biểu đồ Use case của ứng dụng

**Đặc tả Usecase**

1. Đọc hướng dẫn

|  |  |
| --- | --- |
| Usecase Nội dung | |
| Tên Usecase | Đọc hướng dẫn |
| Mô tả | Usecase cho phép người dùng thực hiện chức năng đọc hướng dẫn để nắm rõ luật chơi |
| Actor | Người dùng |
| Điều kiện kích hoạt | Khi người dùng chọn chức năng “Hướng dẫn” trên tab menu |
| Tiền điều kiện | Người dùng đã tải ứng dụng về máy |
| Luồng sự kiện chính | 1.Người dùng truy cập thành công vào ứng dụng  2.Người dùng chọn tab hướng dẫn trên tab menu |
|  |  |

Bảng 1: Đặc tả use case Hướng dẫn

1. Bói tình duyên theo tên

|  |  |
| --- | --- |
| Usecase Nội dung | |
| Tên Usecase | Bói tình duyên theo tên |
| Mô tả | Usecase cho phép người dùng thực hiện chức năng bói tình duyên theo tên của cặp đôi |
| Actor | Người dùng |
| Điều kiện kích hoạt | Khi người dùng chọn chức năng “Bói tên” của ứng dụng |
| Tiền điều kiện | Đã tải ứng dụng thành công về máy |
| Luồng sự kiện chính | Người dùng nhập tên của người thứ nhất  Người dùng nhập tên của người thứ hai  Người dùng nhấn “Xem kết quả” để xem kết quả bói tình duyên theo tên. |

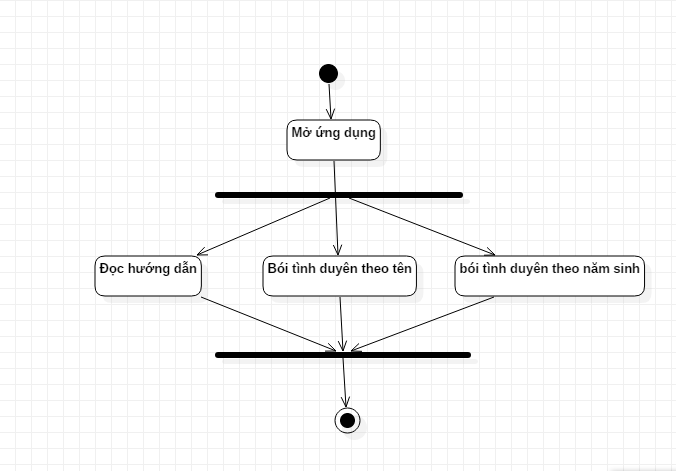
Bảng 2: Đặc tả Use case Bói theo tên

1. Bói tình duyên theo năm sinh

|  |  |
| --- | --- |
| Usecase Nội dung | |
| Tên Usecase | Bói tình duyên theo năm sinh |
| Mô tả | Usecase cho phép người dùng thực hiện chức năng bói tình duyên theo năm sinh |
| Actor | Người dùng |
| Điều kiện kích hoạt | Khi người dùng chọn chức năng “Bói tuổi” của ứng dụng |
| Tiền điều kiện | Người dùng tải thành công ứng dụng về máy |
| Luồng sự kiện chính | Từ giao diện chính của hệ thống, người dùng chọn tab “Bói tuổi” trên tab menu  Người dùng nhập năm sinh của người thứ nhất  Người dùng nhập năm sinh của người thứ hai  Người dùng bấm chọn “Xem kết quả” để xem kết quả bói theo năm sinh của hai người |

Bảng 3: Đặc tả Use case bói theo năm sinh

* 1. Biểu đồ hoạt động



Hình 3: Biểu đồ hoạt động của ứng dụng

* 1. Biểu đồ trình tự

Ảnh có chứa văn bản, bản đồ

Mô tả được tạo tự động

Hình 4: Biểu đồ trình tự của ứng dụng

# **Các chức năng**

## Chức năng Hướng dẫn

1. Code và giải thích

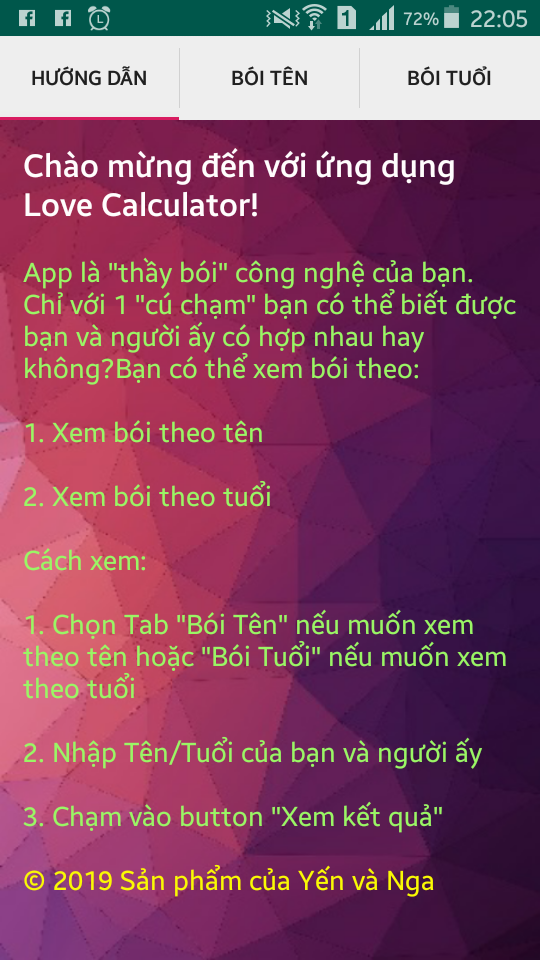
[Love-Calculator](https://github.com/yenbka/Love-Calculator)/[app](https://github.com/yenbka/Love-Calculator/tree/master/app)/[src](https://github.com/yenbka/Love-Calculator/tree/master/app/src)/[main](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main)/[res](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/res)/[layout](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/res/layout)/**help.xml**

|  |
| --- |
|  |
|  | <?xml version="1.0" encoding="utf-8"?>  <LinearLayout |
|  | xmlns:android="http://schemas.android.com/apk/res/android" |
|  | android:orientation="vertical" |
|  | android:layout\_width="match\_parent" |
|  | android:layout\_height="match\_parent" |
|  | android:background="@drawable/help" |
|  | > |
|  |  |
|  | <!-- Screen Design for Photos --> |
|  | <TextView |
|  | android:id="@+id/helpview" |
|  | android:padding="15dip" |
|  | android:textSize="18dip" |
|  | android:layout\_width="fill\_parent" |
|  | android:layout\_height="wrap\_content"/> |
|  |  |
|  | </LinearLayout> |

[Love-Calculator](https://github.com/yenbka/Love-Calculator)/[app](https://github.com/yenbka/Love-Calculator/tree/master/app)/[src](https://github.com/yenbka/Love-Calculator/tree/master/app/src)/[main](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main)/[java](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/java)/[com](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/java/com)/[example](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/java/com/example)/[myapplication](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/java/com/example/myapplication)/**Help.java**

|  |  |
| --- | --- |
|  | package com.example.myapplication;  import android.app.Activity; |
|  | import android.os.Bundle; |
|  | import android.widget.TextView; |
|  |  |
|  | import java.text.BreakIterator; |
|  |  |
|  | public class Help extends Activity { |
|  | @Override |
|  | protected void onCreate(Bundle savedInstanceState) { |
|  | // TODO Auto-generated method stub |
|  | super.onCreate(savedInstanceState); |
|  | setContentView(R.layout.help); |
|  | TextView mytextview; |
|  | mytextview = findViewById(R.id.helpview); |
|  | String htmlcontent = |
|  | "<h4><font color= \"#FFFDFD\">Chào mừng đến với ứng dụng Love Calculator!</font></h4>" |
|  | + "<p><font color=\"#9AF665\">App là \"thầy bói\" công nghệ của bạn. Chỉ với 1 \"cú chạm\" bạn có thể biết được bạn và người ấy có hợp nhau hay không?" + |
|  | "Bạn có thể xem bói theo: </font></p>" |
|  | +"<p><font color=\"#9AF665\">1. Xem bói theo tên</font></p>" |
|  | +"<p><font color=\"#9AF665\">2. Xem bói theo tuổi</font></p>" |
|  | +"<p><font color=\"#9AF665\">Cách xem:</font></p>" + |
|  | "<p><font color=\"#9AF665\">1. Chọn Tab \"Bói Tên\" nếu muốn xem theo tên hoặc \"Bói Tuổi\" nếu muốn xem theo tuổi </font></p>" + |
|  | "<p><font color=\"#9AF665\">2. Nhập Tên/Tuổi của bạn và người ấy</font></p>" + |
|  | "<p><font color=\"#9AF665\">3. Chạm vào button \"Xem kết quả\"</font></p>" |
|  | +"<p><font color=\"#FAFA03\">&copy; 2019 Sản phẩm của Yến và Nga</font><p>"; |
|  | //Nhúng HTML trong textview |
|  | mytextview.setText(android.text.Html.fromHtml(htmlcontent)); |
|  | } |
|  | } |

1. Giao diện minh họa



Hình 5: Giao diện Hướng dẫn

* 1. Chức năng Bói theo tên

1. Code và giải thích

Đầu tiên sẽ tạo file thiết kế giao diện tên là **name.xml**. Trong file này sử dụng LinearLayout để sắp xếp các textview, edittext, button trong màn hình “Bói Tên”.

**Love-Calculator/app/src/main/res/layout/name.xml**

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> |
|  | <LinearLayout |
|  | xmlns:android=<http://schemas.android.com/apk/res/android>  // LinearLayout theo chiều dọc vertical |
|  | android:orientation="vertical"  // chiều rộng của view sẽ bằng đúng chiều rộng của phần tử cha chứa nó. |
|  | android:layout\_width="match\_parent"  // chiều cao của view sẽ bằng đúng chiều cao của phần tử cha chứa nó. |
|  | android:layout\_height="match\_parent"  // set vị trí đặt nội dung của view là ở trung tâm |
|  | android:gravity="center"  // set background là ảnh có tên là name.jpg được đặt trong thư mục drawable |
|  | android:background="@drawable/name"> |
|  |  |
|  | <!-- Screen Design for Name -->  //set TextView cho Text cố định “Tên của bạn” |
|  | <TextView  //chiều rộng của view phụ thuộc vào content của view |
|  | android:layout\_width="wrap\_content"  // chiều cao của view phụ thuộc vào content của view |
|  | android:layout\_height="wrap\_content" |
|  | android:text="Tên của bạn" |
|  | android:gravity="center" |
|  | android:textColor="#FFF"  //set size text là 25sp |
|  | android:textSize="25sp" /> |
|  |  |
|  | <EditText  //set id cho EditText là nameField |
|  | android:id="@+id/nameField" |
|  | android:layout\_width="match\_parent" |
|  | android:layout\_height="wrap\_content" |
|  | android:background="@android:color/transparent" |
|  | android:ems="10"  //Các ký tự được nhập trong EditText |
|  | android:digits="abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ " |
|  | android:gravity="center" |
|  | android:hint="" |
|  | android:inputType="textPersonName" |
|  | android:textColor="#FFF" |
|  | android:textColorHint="@android:color/white" |
|  | android:textSize="20sp" |
|  | android:paddingBottom="20px"/> |
|  | <TextView |
|  | android:layout\_width="wrap\_content" |
|  | android:layout\_height="wrap\_content" |
|  | android:text="Tên của người ấy" |
|  | android:gravity="center" |
|  | android:textColor="#FFF" |
|  | android:textSize="25sp" /> |
|  | <EditText  //set id cho EditText là crushnameField |
|  | android:id="@+id/crushnameField" |
|  | android:layout\_width="match\_parent" |
|  | android:layout\_height="wrap\_content" |
|  | android:background="@android:color/transparent" |
|  | android:ems="10" |
|  | android:gravity="center" |
|  | //set chỉ được phép nhập các ký tự này android:digits="abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ " |
|  | android:hint="" |
|  | android:inputType="textPersonName" |
|  | android:textColor="#FFF" |
|  | android:textColorHint="@android:color/white" |
|  | android:textSize="20sp" |
|  | android:paddingBottom="20px"/> |
|  |  |
|  | <Button  //set ID cho Button là resbtn |
|  | android:id="@+id/resbtn" |
|  | android:layout\_width="wrap\_content" |
|  | android:layout\_height="wrap\_content"  //set vị trí của button là ở trung tâm màn hình |
|  | android:layout\_gravity="center" |
|  | android:background="@android:color/transparent" |
|  | android:text="Xem kết quả" |
|  | android:textColor="@android:color/white"  //set size text cho button là 20sp |
|  | android:textSize="20sp" /> |
|  | <TextView |
|  | android:id="@+id/resview" |
|  | android:layout\_width="match\_parent" |
|  | android:layout\_height="wrap\_content" |
|  | android:gravity="center" |
|  | android:textColor="#FFF" |
|  | android:textSize="20sp" /> |
|  | <TextView |
|  | android:id="@+id/resview1" |
|  | android:layout\_width="match\_parent" |
|  | android:layout\_height="wrap\_content" |
|  | android:gravity="center" |
|  | android:textColor="#FFF" |
|  | android:textSize="20sp" /> |
|  |  |
|  |  |
|  | </LinearLayout> |

**Love-Calculator/app/src/main/res/layout/activity\_main.xml**

**TabHost -** giao diện điều khiển bằng thẻ tab cho phép người dùng có thể chuyển đổi các khung hình khác nhau trong cùng một giao diện Activity

TabHost ở đây điều khiển 3 tab là 3 view trong mỗi Tab là name.xml, age.xml, help.xml

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> |
|  | <TabHost xmlns:android="http://schemas.android.com/apk/res/android" |
|  | xmlns:tools=<http://schemas.android.com/tools>  //set ID của Tabhost là tabhost |
|  | android:id="@android:id/tabhost" |
|  | android:layout\_width="fill\_parent" |
|  | android:layout\_height="fill\_parent" |
|  | android:paddingBottom="0dp" |
|  | android:paddingLeft="0dp" |
|  | android:paddingRight="0dp" |
|  | android:paddingTop="0dp" |
|  | tools:context=".MainActivity" > |
|  |  |
|  | <LinearLayout |
|  | android:orientation="vertical" |
|  | android:layout\_width="fill\_parent" |
|  | android:layout\_height="fill\_parent">  //TabWidget định dạng cho các Tab button |
|  | <TabWidget  //set ID là tabs |
|  | android:id="@android:id/tabs" |
|  | android:layout\_width="fill\_parent" |
|  | android:layout\_height="wrap\_content" />  //FrameLayout – layout chứa nội dung cho các tab |
|  | <FrameLayout  //set ID là tabcontent |
|  | android:id="@android:id/tabcontent" |
|  | android:layout\_width="fill\_parent" |
|  | android:layout\_height="fill\_parent"/> |
|  | </LinearLayout> |
|  | </TabHost> |

[**Love-Calculator**](https://github.com/yenbka/Love-Calculator)**/**[**app**](https://github.com/yenbka/Love-Calculator/tree/master/app)**/**[**src**](https://github.com/yenbka/Love-Calculator/tree/master/app/src)**/**[**main**](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main)**/**[**java**](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/java)**/**[**com**](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/java/com)**/**[**example**](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/java/com/example)**/**[**myapplication**](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/java/com/example/myapplication)**/Name.java**

Name.java là lớp xử lý logic cho View name.xml

|  |
| --- |
| package com.example.myapplication; |
|  |  |
|  | import android.app.Activity; |
|  | import android.os.Bundle; |
|  | import android.support.v7.widget.Toolbar; |
|  | import android.text.Editable; |
|  | import android.view.Menu; |
|  | import android.view.MenuItem; |
|  | import android.view.View; |
|  | import android.widget.Button; |
|  | import android.widget.EditText; |
|  | import android.widget.TextView; |
|  | import android.widget.Toast; |
|  |  |
|  | public class Name extends Activity { |
|  | @Override |
|  | protected void onCreate(Bundle savedInstanceState) { |
|  | super.onCreate(savedInstanceState);  //SetContentView – nạp layout name.xml đã thiết kế |
|  | setContentView(R.layout.name); |
|  |  |
|  |  |
|  | Button cal = (Button) findViewById(R.id.resbtn); |
|  | final TextView res = (TextView) findViewById(R.id.resview); |
|  | final TextView res1 = (TextView) findViewById(R.id.resview1); |
|  | final EditText name = (EditText) findViewById(R.id.nameField); |
|  | final EditText cname = (EditText) findViewById(R.id.crushnameField); |
|  | //Xử lý sự kiện khi click vào button “Xem kết quả” có id đặt là resbtn |
|  | cal.setOnClickListener(new View.OnClickListener() { |
|  | @Override |
|  | public void onClick(View view) { |
|  | res.setText(""); |
|  | res1.setText(""); |
|  | Editable n = name.getText(); |
|  | Editable cn = cname.getText();  // ghép chuỗi ký tự và đổi thành ký tự hoa |
|  | String concat = String.valueOf(n).concat(String.valueOf(cn)).toUpperCase(); |
|  | if ((n.toString().trim().length() == 0) || (cn.toString().trim().length() == 0)) { |
|  | Toast.makeText(Name.this, "Vui lòng không bỏ trống!", Toast.LENGTH\_LONG).show(); |
|  | } else { |
|  | int sum = 0; |
|  | for (int i = 0; i < concat.length(); i++) { |
|  | char character = concat.charAt(i); |
|  | int ascii = (int) character;  // Tổng mã ASCII theo hệ thập phân |
|  | sum += ascii; |
|  | }  //Kết quả sẽ lấy phần dư chia cho 100 |
|  | res.setText("Phần trăm tình duyên giữa " + n + " và " + cn + " là " + sum % 100 + "%"); |
|  | if(sum % 100 > 80){ |
|  | res1.setText("Xin chúc mừng! Tên của 2 bạn sinh ra là dành cho nhau rồi! Còn chần chờ gì nữa mà không tỏ tình ngay thôi nào!"); |
|  | } |
|  | if(sum % 100 < 50){ |
|  | res1.setText("Thật tiếc! Duyên lỡ mất rồi! Hãy tiếp tục tìm một nửa của mình nào!"); |
|  | } |
|  | if(50 <= sum % 100 && sum % 100 <= 80){ |
|  | res1.setText("Hai bạn khá hợp nhau! Hãy cùng cố gắng nhé!"); |
|  | } |
|  |  |
|  | } |
|  | } |
|  | }); |
|  | } |
|  | } |

[**Love-Calculator**](https://github.com/yenbka/Love-Calculator)**/**[**app**](https://github.com/yenbka/Love-Calculator/tree/master/app)**/**[**src**](https://github.com/yenbka/Love-Calculator/tree/master/app/src)**/**[**main**](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main)**/**[**java**](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/java)**/**[**com**](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/java/com)**/**[**example**](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/java/com/example)**/**[**myapplication**](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/java/com/example/myapplication)**/MainActivity.java**

|  |
| --- |
| package com.example.myapplication; |
|  |  |
|  | import android.app.TabActivity; |
|  | import android.content.Intent; |
|  | import android.os.Bundle; |
|  | import android.support.v4.app.FragmentActivity; |
|  | import android.support.v4.app.FragmentTabHost; |
|  | import android.widget.TabHost; |
|  |  |
|  | public class MainActivity extends TabActivity { |
|  | protected void onCreate(Bundle savedInstanceState) { |
|  | super.onCreate(savedInstanceState); |
|  | setContentView(R.layout.activity\_main); |
|  |  |
|  | TabHost tabHost = getTabHost(); |
|  |  |
|  | // Tab for Help |
|  | TabHost.TabSpec photospec = tabHost.newTabSpec("Hướng dẫn"); |
|  | photospec.setIndicator("Hướng dẫn", getResources().getDrawable(R.drawable.help)); |
|  | Intent photosIntent = new Intent(this, Help.class); |
|  | photospec.setContent(photosIntent); |
|  |  |
|  | // Tab for Name |
|  | TabHost.TabSpec songspec = tabHost.newTabSpec("Bói Tên"); |
|  | // setting Title and Icon for the Tab |
|  | songspec.setIndicator("Bói Tên", getResources().getDrawable(R.drawable.help)); |
|  | Intent songsIntent = new Intent(this, Name.class); |
|  | songspec.setContent(songsIntent); |
|  |  |
|  | // Tab for Age |
|  | TabHost.TabSpec videospec = tabHost.newTabSpec("Bói tuổi"); |
|  | videospec.setIndicator("Bói tuổi", getResources().getDrawable(R.drawable.help)); |
|  | Intent videosIntent = new Intent(this, Age.class); |
|  | videospec.setContent(videosIntent); |
|  |  |
|  | // Adding all TabSpec to TabHost |
|  | tabHost.addTab(photospec); // Adding help tab |
|  | tabHost.addTab(songspec); // Adding name tab |
|  | tabHost.addTab(videospec); // Adding age tab |
|  | } |
|  | } |

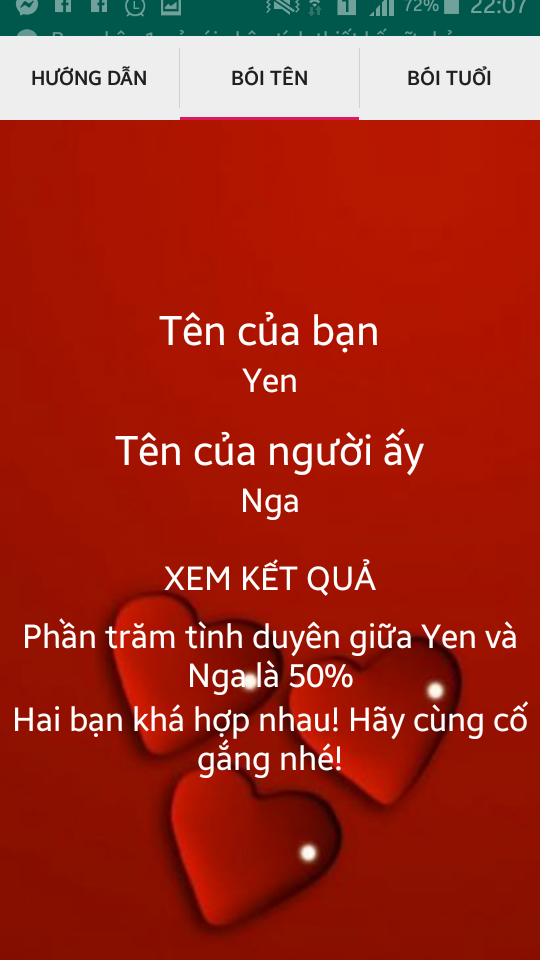
[**Love-Calculator**](https://github.com/yenbka/Love-Calculator)**/**[**app**](https://github.com/yenbka/Love-Calculator/tree/master/app)**/[src](https://github.com/yenbka/Love-Calculator/tree/master/app/src)/**[**main**](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main)**/AndroidManifest.xml**

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> |
|  | <manifest xmlns:android="http://schemas.android.com/apk/res/android" |
|  | package="com.example.myapplication"> |
|  |  |
|  | <application |
|  | android:allowBackup="true" |
|  | android:icon="@mipmap/icon"  //set tên của app |
|  | android:label="@string/app\_name" |
|  | android:roundIcon="@mipmap/ic\_launcher\_round" |
|  | android:supportsRtl="true" |
|  | android:theme="@style/AppTheme"> |
|  | <activity android:name=".MainActivity"> |
|  | <intent-filter> |
|  | <action android:name="android.intent.action.MAIN" /> |
|  |  |
|  | <category android:name="android.intent.category.LAUNCHER" /> |
|  | </intent-filter> |
|  | </activity> |
|  |  |
|  | <!-- Name Activity --> |
|  | <activity android:name=".Name" /> |
|  |  |
|  | <!-- Help Activity --> |
|  | <activity android:name=".Help" /> |
|  |  |
|  | <!-- Age Activity 🡪 |
|  | <activity android:name=”.Age” /> |
|  | </application> |
|  |  |
|  | </manifest> |

[**Love-Calculator**](https://github.com/yenbka/Love-Calculator)**/**[**app**](https://github.com/yenbka/Love-Calculator/tree/master/app)**/[src](https://github.com/yenbka/Love-Calculator/tree/master/app/src)/**[**main**](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main)**/**[**res**](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/res)**/**[**values**](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/res/values)**/strings.xml**

|  |
| --- |
| <resources> |
| //Set tên cho App |
|  | <string name="app\_name">Love Calculator</string> |
|  | </resources> |

1. Giao diện minh họa



Hình 6: Giao diện bói theo tên

* 1. Chức năng Bói theo năm sinh

1. Code và giải thích

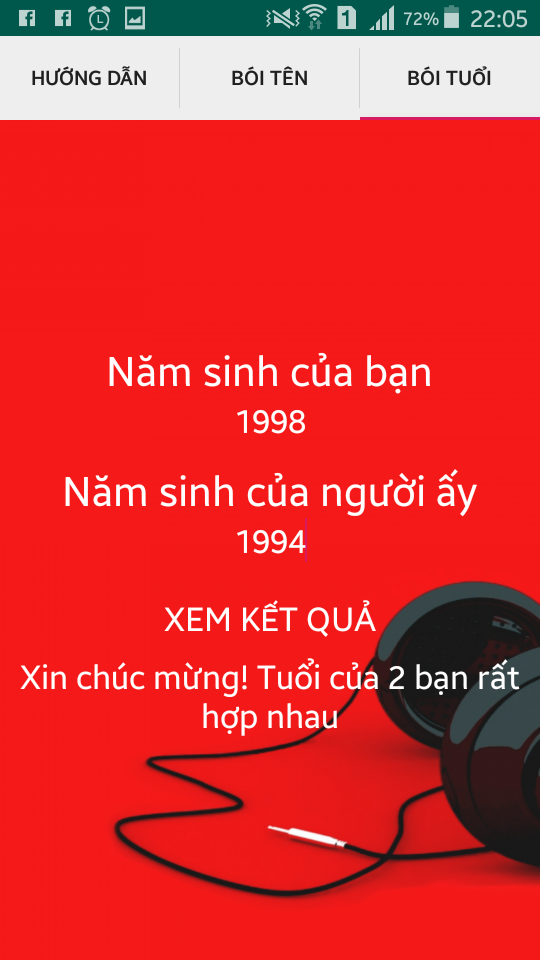
[Love-Calculator](https://github.com/yenbka/Love-Calculator)/[app](https://github.com/yenbka/Love-Calculator/tree/master/app)/[src](https://github.com/yenbka/Love-Calculator/tree/master/app/src)/[main](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main)/[res](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/res)/[layout](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/res/layout)/**age.xml**

|  |  |
| --- | --- |
|  | <?xml version="1.0" encoding="utf-8"?><LinearLayout |
|  | xmlns:android="http://schemas.android.com/apk/res/android" |
|  | android:orientation="vertical" |
|  | android:layout\_width="match\_parent" |
|  | android:layout\_height="match\_parent" |
|  | android:gravity="center" |
|  | android:background="@drawable/age"> |
|  |  |
|  | <!-- Screen Design for Name --> |
|  | <TextView |
|  | android:layout\_width="wrap\_content" |
|  | android:layout\_height="wrap\_content" |
|  | android:text="Năm sinh của bạn" |
|  | android:gravity="center" |
|  | android:textColor="#FFF" |
|  | android:textSize="25sp" /> |
|  |  |
|  | <EditText |
|  | android:id="@+id/ageField" |
|  | android:layout\_width="match\_parent" |
|  | android:layout\_height="wrap\_content" |
|  | android:background="@android:color/transparent" |
|  | android:ems="10" |
|  | android:digits="1234567890"  <!—Chỉ hiển thị các nút số trên bàn phím mềm Android --> |
|  | android:gravity="center" |
|  | android:hint="" |
|  | android:inputType="textPersonName" |
|  | android:textColor="#FFF" |
|  | android:textColorHint="@android:color/white" |
|  | android:textSize="20sp" |
|  | android:paddingBottom="20px"/> |
|  | <TextView |
|  | android:layout\_width="wrap\_content" |
|  | android:layout\_height="wrap\_content" |
|  | android:text="Năm sinh của người ấy" |
|  | android:gravity="center" |
|  | android:textColor="#FFF" |
|  | android:textSize="25sp" /> |
|  | <EditText |
|  | android:id="@+id/crushageField" |
|  | android:layout\_width="match\_parent" |
|  | android:layout\_height="wrap\_content" |
|  | android:background="@android:color/transparent" |
|  | android:ems="10" |
|  | android:gravity="center" |
|  | android:digits="1234567890" |
|  | android:hint="" |
|  | android:inputType="textPersonName" |
|  | android:textColor="#FFF" |
|  | android:textColorHint="@android:color/white" |
|  | android:textSize="20sp" |
|  | android:paddingBottom="20px"/> |
|  |  |
|  | <Button |
|  | android:id="@+id/agebtn" |
|  | android:layout\_width="wrap\_content" |
|  | android:layout\_height="wrap\_content" |
|  | android:layout\_gravity="center" |
|  | android:background="@android:color/transparent" |
|  | android:text="Xem kết quả" |
|  | android:textColor="@android:color/white" |
|  | android:textSize="20sp" /> |
|  |  |
|  | <TextView |
|  | android:id="@+id/ageview" |
|  | android:layout\_width="match\_parent" |
|  | android:layout\_height="wrap\_content" |
|  | android:gravity="center" |
|  | android:textColor="#FFF" |
|  | android:textSize="20sp" /> |
|  | </LinearLayout> |

[Love-Calculator](https://github.com/yenbka/Love-Calculator)/[app](https://github.com/yenbka/Love-Calculator/tree/master/app)/[src](https://github.com/yenbka/Love-Calculator/tree/master/app/src)/[main](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main)/[java](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/java)/[com](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/java/com)/[example](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/java/com/example)/[myapplication](https://github.com/yenbka/Love-Calculator/tree/master/app/src/main/java/com/example/myapplication)/**Age.java**

|  |  |
| --- | --- |
|  | package com.example.myapplication;  import android.app.Activity; |
|  | import android.os.Bundle; |
|  | import android.text.Editable; |
|  | import android.view.View; |
|  | import android.widget.Button; |
|  | import android.widget.EditText; |
|  | import android.widget.TextView; |
|  | import android.widget.Toast; |
|  |  |
|  | public class Age extends Activity { |
|  | @Override |
|  | protected void onCreate(Bundle savedInstanceState) { |
|  | super.onCreate(savedInstanceState); |
|  | setContentView(R.layout.age); |
|  |  |
|  |  |
|  | Button cal = (Button) findViewById(R.id.agebtn); |
|  | final TextView res1 = (TextView) findViewById(R.id.ageview); |
|  | final EditText age = (EditText) findViewById(R.id.ageField); |
|  | final EditText cage = (EditText) findViewById(R.id.crushageField); |
|  |  |
|  | cal.setOnClickListener(new View.OnClickListener() { |
|  | @Override |
|  | public void onClick(View view) { |
|  | res1.setText(""); |
|  | Editable n = age.getText(); |
|  | Editable ca = cage.getText(); |
|  | if ((n.toString().trim().length() == 0) || (ca.toString().trim().length() == 0)) {  //Kiểm tra dữ liệu nhập vào, không được để tr |
|  | Toast.makeText(Age.this, "Vui lòng không bỏ trống!", Toast.LENGTH\_LONG).show(); |
|  | } else { |
|  | String so1 = age.getText().toString(); |
|  | int year1 = Integer.parseInt(so1); |
|  |  |
|  | String so2 = cage.getText().toString(); |
|  | int year2 = Integer.parseInt(so2); |
|  |  |
|  | int hieu1 = year2 - year1; |
|  | int hieu2 = year1 - year2; |
|  | //Tính khoảng cách tuổi giữa hai người |
|  | if (hieu1 == 4 || hieu1 == 2 || hieu1 == 8 || hieu2 == 4 || hieu2 == 2 || hieu2 == 8) { |
|  | res1.setText("Xin chúc mừng! Tuổi của 2 bạn rất hợp nhau"); |
|  | } |
|  | if (hieu1 == 3 || hieu1 == 5 || hieu1 == 7 || hieu1 == 9 || hieu2 == 3 || hieu2 == 5 || hieu2 == 7 || hieu2 == 9) { |
|  | res1.setText("Thật tiếc! Tuổi của 2 bạn không hợp nhau!"); |
|  | } |
|  | if (hieu1 == 1 || hieu1 == 6 || hieu2 == 1 || hieu2 == 6) { |
|  | res1.setText("Tuổi của 2 bạn khá hợp nhau! Hãy cùng cố gắng nhé!"); |
|  | } |
|  | if (hieu1 >= 10 || hieu2 >= 10) { |
|  | res1.setText("Khoảng cách tuổi là khá lớn! Nhưng cuộc sống lại rất bình yên, đừng ngại ngùng khoảng cách tuổi tác nhé!"); |
|  | } |
|  | } |
|  |  |
|  | } |
|  | }); |
|  | } |
|  | } |

1. Giao diện minh họa



Hình 7: Giao diện bói theo Năm sinh