

Lab 5. Giao diện và ứng dụng (4 tiết)

I. Yêu cầu

- Sinh viên sử dụng công cụ Android Studio để làm bài. Kết quả bài làm cần được chụp lại và giữ lại toàn bộ dự án để sử dụng khi cần thiết.
- Mỗi người làm trên một dự án khác nhau. Mỗi sinh viên chỉ làm trên một dự án
- Khi có yêu cầu, sinh viên nộp qua email giáo viên hoặc một kênh khác.

II. Luyện tập

Luyện tập 1: Xử lý animation trong layout

1. Phạm vi kiến thức vận dụng

- Các kiến thức căn bản về Design UI
 - Drawable
 - Color Schemes
 - String constants
 - Animation
- Các đối tượng View System
 - TextView, ImageView
 - Các ViewGroup gồm:
 - LinearLayout, FrameLayout, TableRow
- Các đối tượng logic
 - Activity

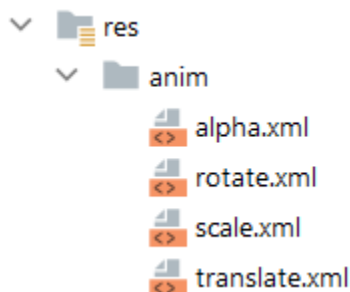
2. Nội dung bài thực hành

Yêu cầu: Tạo 1 ứng dụng **Animation** và xây dựng giao diện layout sau



Gợi ý: Các kiến thức sử dụng

- Tạo 1 layout **xml** trong thư mục res/layout
- Sử dụng các ViewSystem:
 - o ImageView
- Sử dụng các ViewGroup:
 - o LinearLayout hoặc FrameLayout
- Dùng tween-animation để tạo các hiệu ứng động cho ảnh



Nội dung file **alpha**:

Mô tả: Đây là 1 giao diện hiển thị danh sách các animation vui nhộn để điều khiển ảnh 1 con vật, người dùng có thể click vào hiệu ứng animation bất kỳ để làm thay đổi trạng thái của ảnh con vật như mờ, co dãn, di chuyển hoặc xoay.

Yêu cầu giao diện:

- Khối nội dung bên trong layout sẽ có 1 khoảng cách với các cạnh của layout.
- Có 4 button animation trên mỗi hàng và

Yêu cầu về logic:

- Sử dụng kiến thức animation để thực thi hiệu ứng cho ảnh
- Khi người dùng click vào các button hiệu ứng, các animation tương ứng sẽ tác động lên ảnh con vật lặp đi lặp lại.

```
<?xml version="1.0" encoding="utf-8"?>
<alpha xmlns:android="http://schemas.android.com/apk/res/android"
    android:duration="1000"
    android:fromAlpha="1.0"
    android:repeatCount="infinite"
    android:repeatMode="reverse"
    android:toAlpha="0.0" />
```

Nội dung file **scale**:

```
<?xml version="1.0" encoding="utf-8"?>
<scale xmlns:android="http://schemas.android.com/apk/res/android"
    android:duration="1000"
    android:fromXScale="1"
    android:fromYScale="1"
    android:pivotX="50%"
    android:pivotY="50%"
    android:repeatCount="infinite"
    android:repeatMode="reverse"
    android:toXScale="2"
    android:toYScale="2" />
```

Nội dung file **translate**:

```
<?xml version="1.0" encoding="utf-8"?>
<translate xmlns:android="http://schemas.android.com/apk/res/android"
    android:duration="1000"
    android:fromXDelta="0"
    android:repeatCount="infinite"
    android:repeatMode="reverse"
    android:toXDelta="200" />
```

Nội dung file **rotate**:

```
<?xml version="1.0" encoding="utf-8"?>
<rotate xmlns:android="http://schemas.android.com/apk/res/android"
    android:duration="1000"
    android:fromDegrees="0"
    android:pivotX="50%"
    android:pivotY="50%"
    android:repeatCount="infinite"
    android:repeatMode="reverse"
    android:toDegrees="360" />
```

- Download ảnh animal và đưa vào trong thư mục drawable tại link sau:
<https://findicons.com/search/animal/3>

3. Hướng dẫn

Bước 1: Chuẩn bị tài nguyên **drawable**, **string**, **animation** như sau:



Nội dung file **res/values/strings.xml**:

```
<resources>
  <string name="app_name">Animation</string>
  <string name="txt_scale">Scale</string>
  <string name="txt_trans">Trans</string>
  <string name="txt_rotate">Rotate</string>
  <string name="txt_alpha">Alpha</string>
</resources>
```

Bước 2: Tạo file **res/layout/activity_main.xml** có nội dung sau:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  android:gravity="center_horizontal"
  android:orientation="vertical"
  android:padding="20dp"
  tools:context=".MainActivity">
  <ImageView
    android:id="@+id/iv_animal"
    android:layout_width="128dp"
    android:layout_height="128dp"
    android:layout_marginTop="50dp"
    android:src="@drawable/ic_elephant"
    tools:ignore="ContentDescription" />
  <TableRow
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="200dp">

    <Button
      android:id="@+id/bt_alpha"
      style="?android:attr/buttonBarStyle"
      android:layout_width="0dp"
      android:layout_height="wrap_content"
      android:layout_weight="1"
```

```
        android:width="0dp"
        android:backgroundTint="@color/design_default_color_primary"
        android:text="@string/txt_alpha"
        android:textAllCaps="true"
        android:textColor="@color/white"
        android:textSize="14sp" />
    <Button
        android:id="@+id/bt_scale"
        style="?android:attr/buttonBarStyle"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="5dp"
        android:layout_weight="1"
        android:width="0dp"
        android:backgroundTint="@color/design_default_color_primary"
        android:text="@string/txt_scale"
        android:textAllCaps="true"
        android:textSize="14sp" />
    <Button
        android:id="@+id/bt_trans"
        style="?android:attr/buttonBarStyle"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="5dp"
        android:layout_weight="1"
        android:width="0dp"
        android:backgroundTint="@color/design_default_color_primary"
        android:text="@string/txt_trans"
        android:textAllCaps="true"
        android:textSize="14sp" />
    <Button
        android:id="@+id/bt_rotate"
        style="?android:attr/buttonBarStyle"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="5dp"
        android:layout_weight="1"
        android:width="0dp"
        android:backgroundTint="@color/design_default_color_primary"
        android:text="@string/txt_rotate"
        android:textAllCaps="true"
        android:textSize="14sp" />
</TableRow>
</LinearLayout>
```

Bước 3: Triển khai code logic trong activity **MainActivity**

```
package com.techja.animation;
import android.os.Bundle;
import android.view.View;
import android.view.animation.AnimationUtils;
import android.widget.ImageView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity implements View.OnClickListener {
    private ImageView ivAnimal;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        initView();
    }
    private void initView() {
        ivAnimal = findViewById(R.id.iv_animal);
        findViewById(R.id.bt_alpha).setOnClickListener(this);
        findViewById(R.id.bt_rotate).setOnClickListener(this);
        findViewById(R.id.bt_scale).setOnClickListener(this);
        findViewById(R.id.bt_trans).setOnClickListener(this);
    }
    @Override
    public void onClick(View v) {
        if (v.getId() == R.id.bt_alpha) {
            ivAnimal.startAnimation(AnimationUtils.loadAnimation(this, R.anim.alpha));
        } else if (v.getId() == R.id.bt_rotate) {
            ivAnimal.startAnimation(AnimationUtils.loadAnimation(this, R.anim.rotate));
        } else if (v.getId() == R.id.bt_scale) {
            ivAnimal.startAnimation(AnimationUtils.loadAnimation(this, R.anim.scale));
        } else if (v.getId() == R.id.bt_trans) {
            ivAnimal.startAnimation(AnimationUtils.loadAnimation(this, R.anim.translate));
        }
    }
}
```

4. Lưu ý:

- Phương thức `ivAnimal.startAnimation(AnimationUtils.loadAnimation(this, R.anim.alpha));` giúp thực thi hiệu ứng animation lên ảnh.

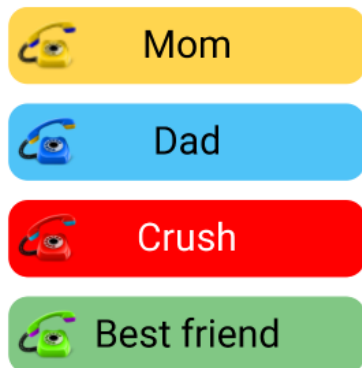
Luyện tập 2 - Xây dựng ứng dụng QuickCall (1)

1. Phạm vi kiến thức vận dụng

- Các kiến thức căn bản về Design UI
 - Drawable
 - Color Schemes
 - String constants
 - Style & themes
 - Layout constraint rules (padding, margin...)
- Các đối tượng View System
 - TextView, ImageView
 - Các ViewGroup gồm:
 - LinearLayout, FrameLayout, TableRow

2. Nội dung bài thực hành

Yêu cầu: Tạo 1 ứng dụng Quick Call và xây dựng giao diện layout sau



Mô tả: Đây là 1 giao diện hiển thị danh sách các danh bạ thân quen và thường xuyên liên lạc nhất của người dùng.

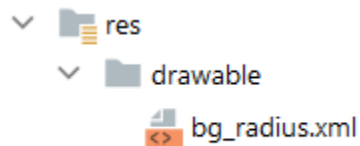
Yêu cầu giao diện:

- Custom thanh actionbar, góc phải có thêm icon dialpad để người dùng nhập số điện thoại
- Khối nội dung bên trong layout sẽ có 1 khoảng cách với các cạnh của layout.
- Phần icon và tên danh bạ phải được bọc trong 1 background bo tròn
- Các danh bạ nằm thành từng hàng có kích thước tương đương nhau.

Gợi ý: Các kiến thức sử dụng

- Tạo 1 layout **xml** trong thư mục res/layout
- Sử dụng các ViewSystem:
 - TextView, ImageView
- Sử dụng các ViewGroup:
 - LinearLayout hoặc FrameLayout
 - Thẻ include

- Dùng drawable **Shape** để tạo 1 giao diện background bo góc



Nội dung file:

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
  <solid android:color="#BFF9FF91" />
  <corners android:radius="20dp" />
</shape>
```

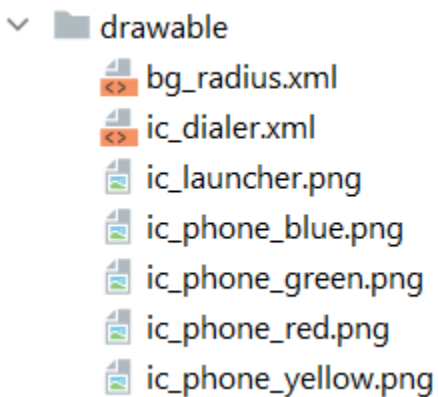
- Sử dụng bộ icon phone sau:



- Lấy ảnh icon dialpad trong kho ảnh vector assets của AndroidStudio

3. Hướng dẫn

Bước 1: Chuẩn bị tài nguyên **drawable**, **string**, **themes** như sau:



Nội dung file **res/values/strings.xml**

```
<resources>
  <string name="app_name">Quick Call</string>
  <string name="txt_mom">Mom</string>
  <string name="txt_dad">Dad</string>
  <string name="txt_crush">Crush</string>
  <string name="txt_best_friend">Best friend</string>
  <string name="phone_mom">096666543</string>
  <string name="phone_dad">096666544</string>
  <string name="phone_crush">096886543</string>
  <string name="phone_best_friend">099966543</string>
</resources>
```

Nội dung file **res/values/themes.xml**

```
<resources xmlns:tools="http://schemas.android.com/tools">
  <!-- Base application theme. -->
  <style name="Theme.Demo" parent="Theme.MaterialComponents.DayNight.NoActionBar">
    <!-- Primary brand color. -->
    <item name="colorPrimary">@color/purple_500</item>
    <item name="colorPrimaryVariant">@color/purple_700</item>
    <item name="colorOnPrimary">@color/white</item>
    <!-- Secondary brand color. -->
    <item name="colorSecondary">@color/teal_200</item>
    <item name="colorSecondaryVariant">@color/teal_700</item>
    <item name="colorOnSecondary">@color/black</item>
    <!-- Status bar color. -->
    <item name="android:statusBarColor" tools:targetApi="l">?attr/colorPrimaryVariant</item>
    <!-- Customize your theme here. -->
  </style>
</resources>
```

Bước 2: Tạo file **res/layout/actionbar_home.xml** có nội dung sau:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:app="http://schemas.android.com/apk/res-auto"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="wrap_content"
  android:background="@color/design_default_color_primary"
  android:orientation="horizontal">

  <TextView
```

```

        android:layout_width="0dp"
        android:layout_height="56dp"
        android:layout_weight="1"
        android:gravity="center_vertical"
        android:paddingStart="16dp"
        android:text="@string/app_name"
        android:textColor="@color/white"
        android:textSize="24sp"
        tools:ignore="RtlSymmetry" />

<ImageView
    android:layout_width="60dp"
    android:layout_height="match_parent"
    android:padding="10dp"
    android:layout_marginEnd="10dp"
    android:src="@drawable/ic_dialer"
    app:tint="@color/white"
    tools:ignore="ContentDescription" />
</LinearLayout>

```

Bước 3: Tạo file **res/layout/activity_main.xml** có nội dung sau:

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <include layout="@layout/actionbar_home" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center_vertical"
        android:orientation="vertical"
        android:padding="20dp">

        <FrameLayout
            android:id="@+id/fr_mom"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@drawable/bg_radius"
            android:backgroundTint="#FFD54F"

```

```
        android:gravity="center_vertical"
        android:padding="10dp"
        android:tag="@string/phone_mom">

        <ImageView
            android:layout_width="60dp"
            android:layout_height="60dp"
            android:src="@drawable/ic_phone_yellow"
            tools:ignore="ContentDescription" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:text="@string/txt_mom"
            android:textColor="@color/black"
            android:textSize="40sp" />
    </FrameLayout>

    <FrameLayout
        android:id="@+id/fr_dad"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:background="@drawable/bg_radius"
        android:backgroundTint="#4FC3F7"
        android:gravity="center_vertical"
        android:padding="10dp"
        android:tag="@string/phone_dad">

        <ImageView
            android:layout_width="60dp"
            android:layout_height="60dp"
            android:src="@drawable/ic_phone_blue"
            tools:ignore="ContentDescription" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:text="@string/txt_dad"
            android:textColor="@color/black"
            android:textSize="40sp" />
    </FrameLayout>
```

```
<FrameLayout
    android:id="@+id/fr_crush"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:background="@drawable/bg_radius"
    android:gravity="center_vertical"
    android:padding="10dp"
    android:tag="@string/phone_crush">

    <ImageView
        android:layout_width="60dp"
        android:layout_height="60dp"
        android:src="@drawable/ic_phone_red"
        tools:ignore="ContentDescription" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:text="@string/txt_crush"
        android:textColor="@color/white"
        android:textSize="40sp" />
</FrameLayout>

<FrameLayout
    android:id="@+id/fr_best_friend"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:background="@drawable/bg_radius"
    android:backgroundTint="#81C784"
    android:gravity="center_vertical"
    android:padding="10dp"
    android:tag="@string/phone_best_friend">

    <ImageView
        android:layout_width="60dp"
        android:layout_height="60dp"
        android:src="@drawable/ic_phone_green"
        tools:ignore="ContentDescription" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
```

```
        android:gravity="center"
        android:text="@string/txt_best_friend"
        android:textColor="@color/black"
        android:textSize="40sp" />
    </FrameLayout>
</LinearLayout>
</LinearLayout>
```

4. Lưu ý:

- Sử dụng style `Theme.MaterialComponents.DayNight.NoActionBar` để ẩn actionBar của layout
- Sử dụng thẻ **include** để thêm actionBar cho layout main
- Mỗi 1 item danh bạ View đã được đính kèm dữ liệu của 1 số điện thoại bằng thuộc tính **tag**.

Luyện tập 2: Xây dựng ứng dụng QuickCall (2)

1. Phạm vi kiến thức vận dụng

- Tiếp nối Phần 1 - Tạo giao diện Quick Call
- Các kiến thức về logic
 - Activity
 - OnClickListener
 - Intent
 - User permission

2. Nội dung bài thực hành

Yêu cầu: Tạo 1 ứng dụng Quick Call đã có giao diện bên dưới và xử lý logic sau.



Mô tả: Đây là 1 giao diện hiển thị danh sách các danh bạ thân quen và thường xuyên liên lạc nhất của người dùng.

Yêu cầu logic:

- Người dùng click vào các item danh bạ để thực hiện cuộc gọi.
- Người dùng click vào icon dialpad để mở màn hình quay số điện thoại



Gợi ý: Các kiến thức sử dụng

- Tạo 1 layout **xml** trong thư mục res/layout
- Sử dụng các ViewSystem:
 - TextView, ImageView
- Sử dụng các ViewGroup:
 - LinearLayout hoặc FrameLayout
 - Thẻ include

2. Hướng dẫn triển khai code logic trong activity

Bước 1: Khai báo quyền người dùng trong Android Manifest

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.techja.demo">

    <uses-permission android:name="android.permission.CALL_PHONE" />

    <application
        android:allowBackup="true"
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:supportRtl="true"
        android:theme="@style/Theme.Demo">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

Bước 2: Xử lý logic trong file MainActivity như sau:

- Ánh xạ các View trong layout ra môi trường code
- Xử lý click vào các item Danh Bạ đồng thời kiểm tra quyền **thực hiện cuộc gọi** xem người dùng đã cấp chưa
- Xử lý click vào icon dial để thực hiện mở màn hình quay số điện thoại

```
package com.techja.demo;

import android.Manifest;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.view.animation.AnimationUtils;
import android.widget.FrameLayout;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;
```

```

public class MainActivity extends AppCompatActivity implements View.OnClickListener {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        initView();
    }
    private void initView() {
        findViewById(R.id.fr_mom).setOnClickListener(this);
        findViewById(R.id.fr_dad).setOnClickListener(this);
        findViewById(R.id.fr_crush).setOnClickListener(this);
        findViewById(R.id.fr_best_friend).setOnClickListener(this);
        findViewById(R.id.iv_dialer).setOnClickListener(this);
    }
    @Override
    public void onClick(View v) {
        v.startAnimation(AnimationUtils.loadAnimation(this,
        androidx.appcompat.R.anim.abc_popup_enter));
        if (v instanceof FrameLayout) {
            processCall((String) v.getTag());
            return;
        }
        gotoDialPad();
    }

    private void gotoDialPad() {
        Intent intent = new Intent(Intent.ACTION_DIAL);
        startActivity(intent);
    }

    private void processCall(String phone) {
        if (checkSelfPermission(Manifest.permission.CALL_PHONE)
            != PackageManager.PERMISSION_GRANTED) {
            requestPermissions(new String[]{Manifest.permission.CALL_PHONE}, 101);
            Toast.makeText(this, "Hãy thực hiện lại sau khi cấp quyền!", Toast.LENGTH_SHORT).show();
            return;
        }
        Intent intent = new Intent(Intent.ACTION_CALL);
        intent.setData(Uri.parse("tel: " + phone));
        startActivity(intent);
    }
}

```


4. Lưu ý:

- Câu lệnh kiểm tra quyền người dùng xem đã được cấp hay chưa:

```
if (checkSelfPermission(Manifest.permission.CALL_PHONE)
    != PackageManager.PERMISSION_GRANTED) {
    requestPermissions(new String[]{Manifest.permission.CALL_PHONE}, 101);
    Toast.makeText(this, "Hãy thực hiện lại sau khi cấp quyền!",
        Toast.LENGTH_SHORT).show();
    return;
}
```

- Câu lệnh để mở ứng dụng Phone nhằm thực hiện cuộc gọi

```
Intent intent = new Intent(Intent.ACTION_CALL);
intent.setData(Uri.parse("tel: " + phone));
startActivity(intent);
```

- Câu lệnh để mở màn hình quay số của ứng dụng Phone

```
Intent intent = new Intent(Intent.ACTION_DIAL);
startActivity(intent);
```

III. Bài tập

Bài tập 1: Trong bài luyện tập 1, thêm nút Random để khi nhấn vào hình con voi sẽ random 1 kiểu animation bất kỳ.

Bài Tập 2: Sinh viên áp dụng các animation trong Luyện tập 1, áp dụng cho các nút trong trong bài Luyện tập 2.

--Hết--