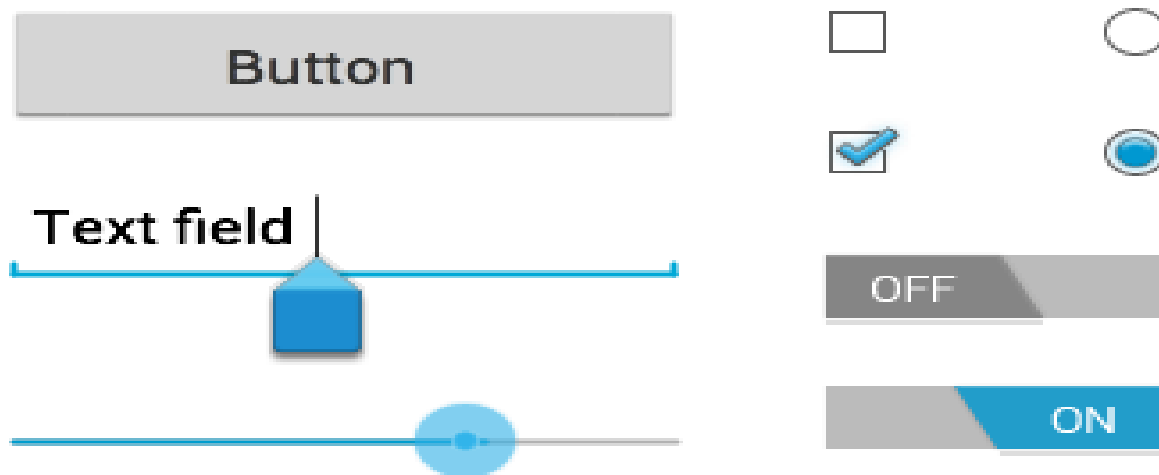




INPUT CONTROL TRONG ANDROID

Khoa Công nghệ Thông tin
Đại học Sư phạm Kỹ thuật TP.HCM

- Input Control là các thành phần có tính tương tác trong giao diện UI của ứng dụng. Android cung cấp nhiều control đa dạng để bạn có thể sử dụng trong UI như button, text field, seek bar, checkbox, zoom button, toggle button, ...



- **TextView** được sử dụng để hiển thị text tới người dùng.

```
<TextView android:id="@+id/text_id"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Hello World" />
```

```
TextView myText = (TextView) findViewById(R.id.text_id);
```

- EditText Là một lớp con được định nghĩa trước của TextView mà bao gồm các khả năng chỉnh sửa đa dạng.

```
<EditText
    android:id="@+id/textview"
    android:layout_height="wrap_content"
    android:layout_width="match_parent"
    android:inputType="text"
    android:maxLines="1"
    android:lines="1"/>
```

Thuộc tính **android:inputType** dùng để thiết lập kiểu nhập liệu, hiện thị

- **AutoCompleteTextView** trong Android hỗ trợ cho người dùng những gợi ý liên quan khi bạn nhập vào trường EditText.

```

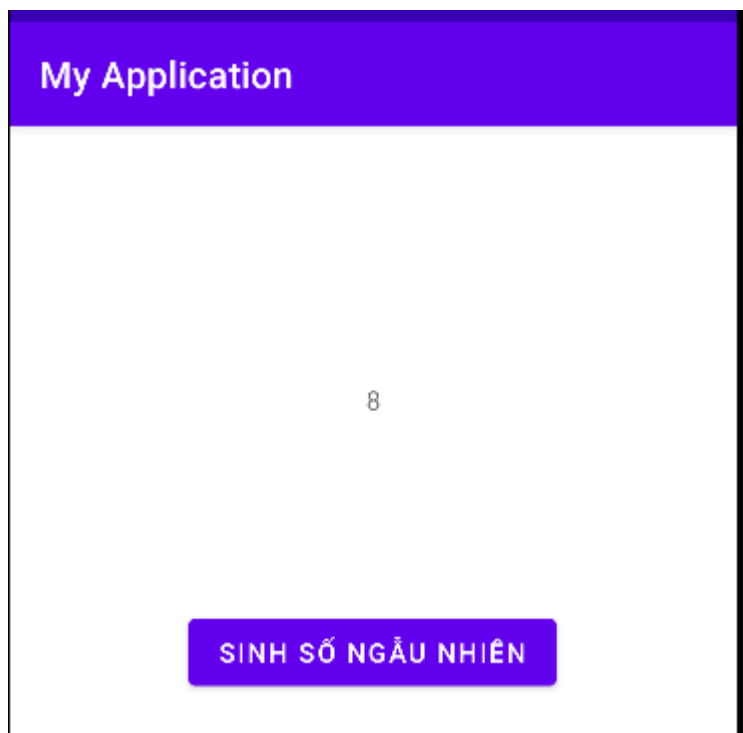
java.lang.Object
└─ android.view.View
    └─ android.widget.TextView
        └─ android.widget.EditText
            └─ android.widget.AutoCompleteTextView
    
```

```

<AutoCompleteTextView
android:id="@+id/autoCompleteTextView1"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignParentLeft="true"
android:layout_below="@+id/textView1"
android:layout_marginLeft="36dp"
android:layout_marginTop="17dp"
android:ems="10"
android:text="">
<requestFocus />
</AutoCompleteTextView>
    
```

Control	Công dụng
Button	Một nút có thể được nhấn, hoặc click bởi người dùng để thực hiện một hành động.
ImageButton	Là một <code>AbsoluteLayout</code> cho bạn khả năng xác định vị trí chính xác của các view con
CheckBox	On/Off có thể được chuyển đổi bởi người dùng.
ToggleButton	Hiển thị trạng thái checked/unchecked giống một nút on/off với một light indicator

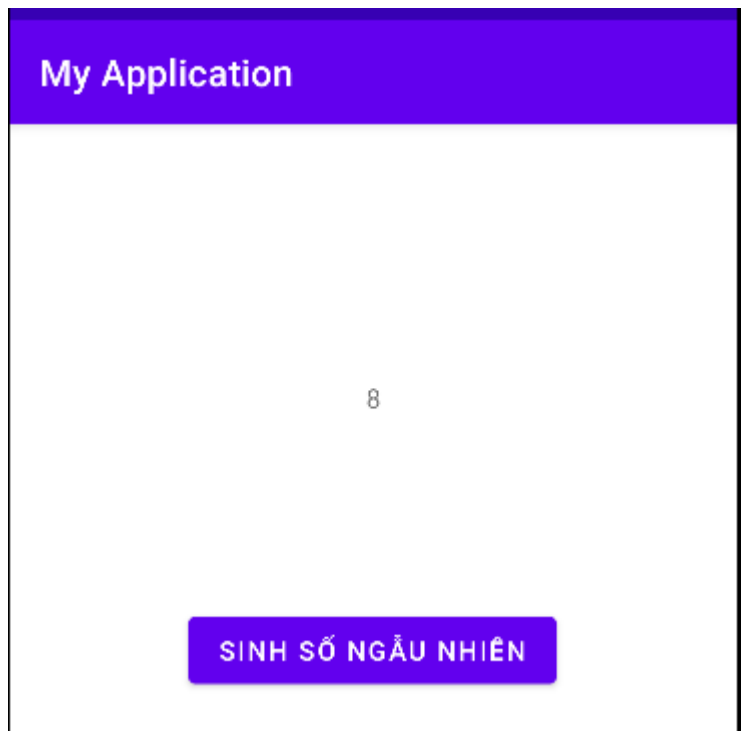
Control	Công dụng
ProgressBar	Cung cấp một phản hồi có thể nhìn thấy về một số tác vụ, như khi chúng ta thực hiện tác vụ ra ngoài trong background.
TimePicker	Cho phép người sử dụng lựa chọn thời gian của một ngày: hoặc chế độ 24 h hoặc chế độ AM/PM
DatePicker	Cho phép người dùng lựa chọn một date



```
<TextView
    android:id="@+id/textViewSo"
    android:layout_width="match_parent"
    android:layout_height="40dp"
    android:textAlignment="gravity"
    android:gravity="center"

    android:text="" />
```

```
<Button
    android:id="@+id/buttonRnd"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="92dp"
    android:text="Sinh số ngẫu nhiên"
```

//ánh xạ

```
txtSoN = (TextView)
findViewById(R.id.textViewSo);
btnRnd = (Button)
findViewById(R.id.buttonRnd);
```

//viết code sinh ngẫu nhiên

```
btnRnd.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View view) {
        //tạo số ngẫu nhiên
        Random random = new Random();
        int number = random.nextInt(10);
        txtSoN.setText(number + ""); //number + ""
        ép kiểu
    }
});
```

My Application

My Application

5

Nhập số thứ nhất

3

Nhập số thứ hai

7

SINH SỐ NGẪU NHIÊN

SINH SỐ NGẪU NHIÊN

My Application



<ImageView

```

    android:scaleType="center"
    android:id="@+id/imageView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="92dp"
    android:src="@drawable/kotlin"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

```

```

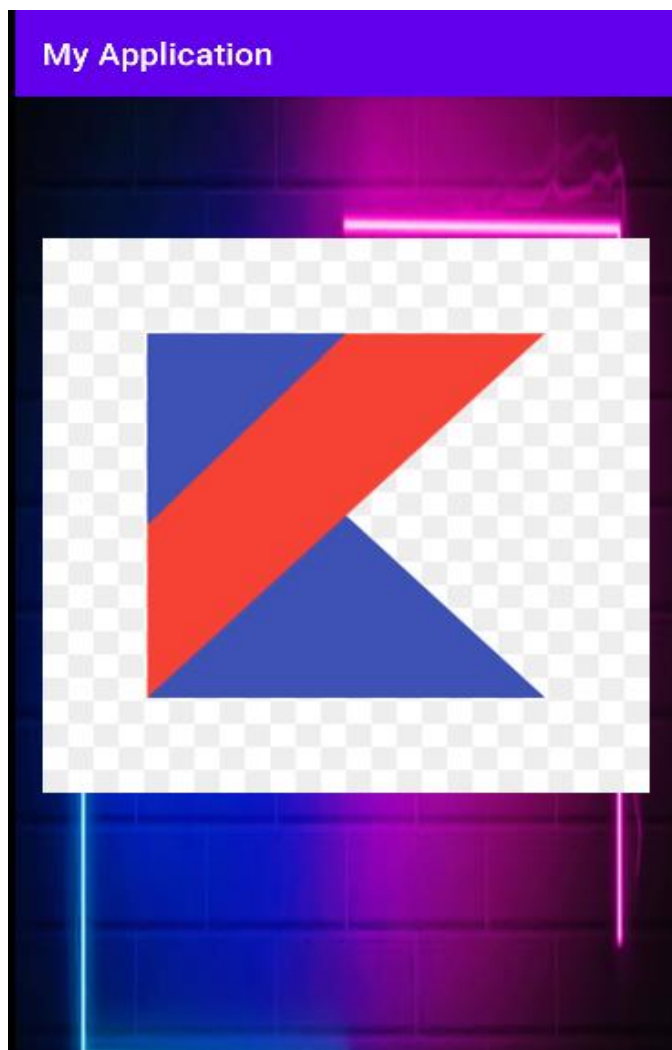
ImageView img1= (ImageView)
findViewById(R.id.imageView1);

```

```

img1.setImageResource(R.drawable.kotlin);

```

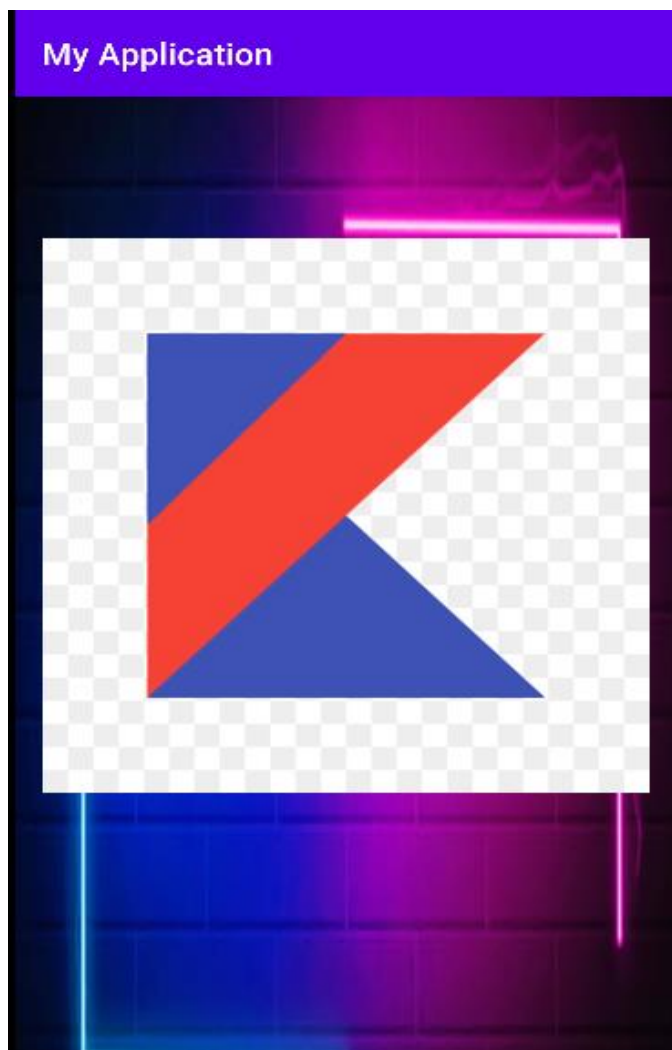


```
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/
android"
    xmlns:app="http://schemas.android.com/apk/res-
auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/bg1"
    android:id="@+id/constraintLayout1"
>
```

```
ConstraintLayout bg = (ConstraintLayout)
findViewById(R.id.constraintLayout1);
```

```
bg.setBackgroundColor(Color.BLUE);
```

```
bg.setBackgroundResource(R.drawable.bg2);
```



Random background

```
ArrayList<Integer> arrayList = new ArrayList<>();
arrayList.add(R.drawable.bg1);
arrayList.add(R.drawable.bg2);
arrayList.add(R.drawable.bg3);
arrayList.add(R.drawable.bg4);
```

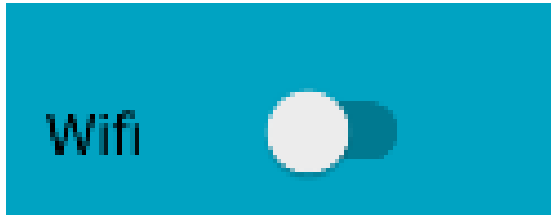
```
Random random = new Random();
int vitri = random.nextInt(arrayList.size());
```

```
bg.setBackgroundResource(arrayList.get(vitri));
```



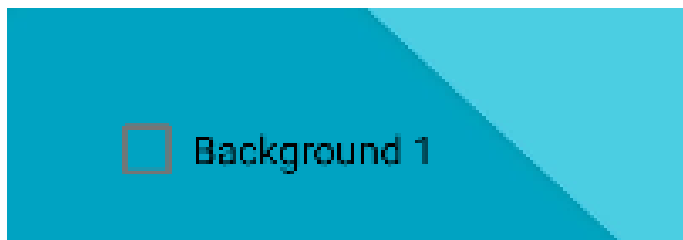
```
<ImageButton
    android:id="@+id/imageButton1"
    android:layout_width="80dp"
    android:layout_height="80dp"
    android:background="@drawable/on"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.45"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.732" />
```

```
ImageButton img2 = (ImageButton)
findViewById(R.id.imageButton1);
img2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        img1.setImageResource(R.drawable.dart);
        img1.getLayoutParams().width=550;
        img1.getLayoutParams().height=550;
    }
});
```



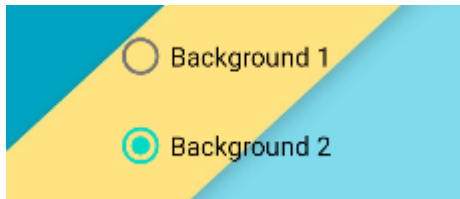
```
<Switch
    android:id="@+id/switch1"
    android:layout_width="97dp"
    android:layout_height="38dp"
    android:text="Wifi"/>
```

```
//switch
Switch sw = (Switch) findViewById(R.id.switch1);
sw.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {
    @Override
    public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {
        if(isChecked){ //isChecked = true
            Toast.makeText(ControlActivity.this, "Wifi đang bật", Toast.LENGTH_LONG).show();
        }else{
            Toast.makeText(ControlActivity.this, "Wifi đang tắt", Toast.LENGTH_LONG).show();
        }
    }
});
```



```
<CheckBox
    android:id="@+id/checkBox"
    android:layout_width="151dp"
    android:layout_height="66dp"
    android:text="Background 1"
```

```
CheckBox ck1 = (CheckBox) findViewById(R.id.checkBox2);
ck1.setOnCheckedChangeListener(new
CompoundButton.OnCheckedChangeListener() {
    @Override
    public void onCheckedChanged(CompoundButton buttonView, boolean isChecked)
    {
        if(isChecked){
            bg.setBackgroundResource(R.drawable.bg3);
        }else{
            bg.setBackgroundResource(R.drawable.bg4);
        }
    }
});
```

//RadioGroup

```
RadioGroup radioGroup = (RadioGroup)
findViewById(R.id.radioGroup1);
radioGroup.setOnCheckedChangeListener(new
RadioGroup.OnCheckedChangeListener() {
    @Override
    public void onCheckedChanged(RadioGroup group,
int checkedId) {
        //checkID trả về ID radio
        switch (checkedId){
            case R.id.radioButton:
                bg.setBackgroundResource(R.drawable.bg3);
                break;
            case R.id.radioButton2:
                bg.setBackgroundResource(R.drawable.bg4);
                break;
        }
    }
});
```

```
<RadioGroup
    android:id="@+id/radioGroup1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" >
    <RadioButton
        android:id="@+id/radioButton"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:checked="true"
        android:text="Background 1" />
    <RadioButton
        android:id="@+id/radioButton2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Background 2" />
</RadioGroup>
```



```
<ProgressBar
    android:id="@+id/progressBar2"
    android:progress="40"
    android:max="100"
    android:secondaryProgress="60"
    style="@android:style/Widget.ProgressBar.Horizontal"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
```

```
//progrebar
```

```
ProgressBar progressBar = (ProgressBar) findViewById(R.id.progressBar2);
img2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        int current = progressBar.getProgress();
        progressBar.setProgress(current + 10);
    }
});
```

```
<ProgressBar
    android:id="@+id/progressBar2"
    android:progress="40"
    android:max="100"
    android:secondaryProgress="60"
    style="@android:style/Widget.ProgressBar.Horizontal"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
```

```
//progrebar
```

```
ProgressBar progressBar = (ProgressBar) findViewById(R.id.progressBar2);
img2.setOnClickListener(new View.OnClickListener() {
```

```
    @Override
```

```
    public void onClick(View v) {
```

```
        int current = progressBar.getProgress();
```

```
        //chạy đều đều >100 quay về 0
```

```
        if (current >= progressBar.getMax()){
```

```
            current=0;
```

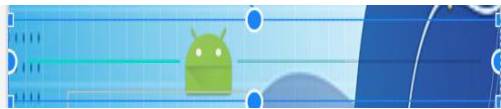
```
        }
```

```
        progressBar.setProgress(current + 10); //thiết lập progress
```

```
    }
```

```
});
```

```
//progrebar
ProgressBar progressBar = (ProgressBar) findViewById(R.id.progressBar2);
img2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        //tự đếm progress
        CountDownTimer countDownTimer = new CountDownTimer(10000,1000) {
            @Override
            public void onTick(long millisUntilFinished) {
                int current = progressBar.getProgress();
                //chạy đều đều >100 quay về 0
                if (current >= progressBar.getMax()){
                    current=0;
                }
                progressBar.setProgress(current + 10); //thiết lập progress
            }
            @Override
            public void onFinish() {
                Toast.makeText(ControlActivity.this, "Hết giờ", Toast.LENGTH_LONG).show();
            }
        };
        countDownTimer.start();
    }
});
```



```
<SeekBar
    android:id="@+id/seekBar"
    style="@android:style/Widget.DeviceDefault.Light.SeekBar"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:max="100"
    android:progress="40"
    android:secondaryProgress="70"
    android:thumb="@android:mipmap/sym_def_app_icon"
```

//SeekBar

```
SeekBar seekBar = (SeekBar) findViewById(R.id.seekBar);
seekBar.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
    @Override
    public void onProgressChanged(SearchBar seekBar, int progress, boolean fromUser) {
        //progress: giá trị của seekbar
        Log.d("AAA", "Giá trị:" + progress);
    }
    @Override
    public void onStartTrackingTouch(SearchBar seekBar) {
        Log.d("AAA", "Start");
    }
    @Override
    public void onStopTrackingTouch(SearchBar seekBar) {
        Log.d("AAA", "Stop");
    }
});
```

- Bước 1: Vẽ button lên ViewGroup
- Bước 2: Tạo Drawable Resource File

```
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="rectangle">
    <solid android:color="#ff00" />
    <size android:height="550dp"
        android:width="200dp"/>
    <corners android:radius="30dp" />
    <gradient android:centerColor="@color/purple_200"/>
</shape>
```

- Bước 3: Gọi Drawable

```
android:background="@drawable/button_custom"
```

- Bước 1: Vẽ button lên ViewGroup
- Bước 2: Tạo Drawable Resource File

```
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="rectangle">
    <solid android:color="#ff00" />
    <size android:height="550dp"
        android:width="200dp"/>
    <corners android:radius="30dp" />
    <gradient android:centerColor="@color/purple_200"/>
</shape>
```

- Bước 3: Gọi Drawable

```
android:background="@drawable/button_custom"
```

- Bước 1: Tạo thư mục menu trong Resources
- Bước 2: Tạo Menu Resource File

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:title="Setting"/>
    <item android:title="Share"/>
    <item android:title="Logout"/>
</menu>
```

- Bước 3: Gọi menu

@Override

```
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu_setting, menu);
    return super.onCreateOptionsMenu(menu);
}
```


□ Bước 1: Bắt sự kiện menu

```
<?xml version="1.0" encoding="utf-8" ?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto">
    <item android:title="Setting"
        android:id="@+id/menuSetting"
        android:icon="@android:drawable/ic_menu_set_as"
        app:showAsAction="always"/>
    <item android:title="Share"
        android:orderInCategory="1"
        android:id="@+id/menuShare"/>
    <item android:title="Logout"
        android:orderInCategory="2"
        android:id="@+id/menuLogout"/>
    <item android:title="Menu con"
        android:orderInCategory="3">
        <menu>
            <item android:title="Menu con 1"/>
            <item android:title="Menu con 2"/>
        </menu>
    </item>
</menu>
```

□ Bước 1: Bắt sự kiện menu

@Override

```
public boolean onOptionsItemSelected(@NonNull MenuItem item) {
    switch (item.getItemId()){
        case R.id.menuSetting:
            //lệnh
            break;
        case R.id.menuShare:
            break;
        case R.id.menuLogout:
            break;
    }

    return super.onOptionsItemSelected(item);
}
```

□ Bước 1: Gọi Popup menu

```
//popup menu
```

```
private void ShowPopupMenu(){
    PopupMenu popupMenu = new PopupMenu(this,btnButton);

    popupMenu.getMenuInflater().inflate(R.menu.menu_setting,popupMenu.getMenu());
    popupMenu.show();
}
```

□ Bước 2: Bắt sự kiện Popup menu

```
//popup menu
private void ShowPopupMenu(){
    PopupMenu popupMenu = new PopupMenu(this, btnButton);
    popupMenu.getMenuInflater().inflate(R.menu.menu_setting, popupMenu.getMenu());
    //bắt sự kiện
    popupMenu.setOnMenuItemClickListener(new PopupMenu.OnMenuItemClickListener() {
        @Override
        public boolean onMenuItemClick(MenuItem item) {
            switch (item.getItemId()){
                case R.id.menuSetting:
                    //lệnh
                    Toast.makeText(ControlActivity2.this, "Bạn đang chọn Setting", Toast.LENGTH_LONG).show();
                    break;
                case R.id.menuShare:
                    btnButton.setText("Chia sẻ");
                    break;
                case R.id.menuLogout:
                    break;
            }
            return false;
        }
    });
    popupMenu.show();
}
```

□ Bước 1: Gọi Context menu

@Override

```
public void onCreateContextMenu(ContextMenu menu, View v,
    ContextMenu.ContextMenuInfo menuInfo) {
    getMenuInflater().inflate(R.menu.menu_setting, menu);
    menu.setHeaderTitle("Context Menu");
    menu.setHeaderIcon(R.mipmap.ic_launcher);
    super.onCreateContextMenu(menu, v, menuInfo);
}
```

```
//đăng ký view cho context menu trong onCreate
registerForContextMenu(btnButton);
```

□ Bước 2: bắt sự kiện Context menu

//bắt sự kiện Context Menu

@Override

```
public boolean onOptionsItemSelected(@NonNull MenuItem item) {
    switch (item.getItemId()){
        case R.id.menuSetting:
            //lệnh
            Toast.makeText(ControlActivity2.this,"Bạn đang chọn
            Setting",Toast.LENGTH_LONG).show();
            break;
        case R.id.menuShare:
            btnButton.setText("Chia sẻ");
            break;
        case R.id.menuLogout:
            break;
    }
    return super.onOptionsItemSelected(item);
}
```

```
private void XacNhanXoa( final int vitri){
    AlertDialog.Builder alert = new AlertDialog.Builder(this);
    alert.setTitle("Thông báo");
    alert.setMessage("Bạn có muốn đăng xuất không");
    alert.setPositiveButton("Có", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            //lệnh nút có
        }
    });
    alert.setNegativeButton("Không", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            //lệnh nút không
        }
    });
    alert.show();
}
```

□ Bước 1: Tạo layout Dialog

Hồ sơ Sinh viên

MSSV

Họ Tên

CẬP NHẬT

```
//tạo hàm Dialog
private void Dialog1(){
    Dialog dialog = new Dialog(this);

    dialog.requestWindowFeature(Window.FEATURE_NO_TITLE);
    dialog setContentView(R.layout.dialog);

    dialog.setCanceledOnTouchOutside(false);
    //ánh xạ
    EditText editText1 = (EditText)
    dialog.findViewById(R.id.editNumber1);
    //viết code sự kiện
    //bắt sự kiện Dialog
    dialog.show(); //hủy gọi dialog.dismiss();
}
```


- 1. Viết hàm tự động thay đổi hình nền của app khi mỗi lần load app.
- 2. Viết hàm thay đổi hình nền app khi bấm vào control Switch.

- Nguyễn Hữu Trung
- 0908617108
- trungnh@hcmute.edu.vn
- utex.hcmute.edu.vn