



## **THIẾT KẾ GIAO DIỆN ANDROID**

### **BÀI 7: MATERIAL DESIGN**

- ⊙ Kết thúc bài học này bạn có khả năng
  - ⊙ Hiểu được 1 số view mới của material design
  - ⊙ Tạo hiệu ứng ẩn hiện thành công cụ khi trước



## Phần I: Material Design, các view

 A Giới thiệu và cài đặt Material Design

 B SnackBar

 C TextInputLayout

 D Percent Layout

 E FloatingActionButton

 F ButtonNavigationView

## Phần II: Các view hỗ trợ ẩn hiện khi cuộn

 G CoordinatorLayout

 H AppBarLayout

 I CollapsingToolBarLayout

 J Phối hợp các view tạo hiệu ứng

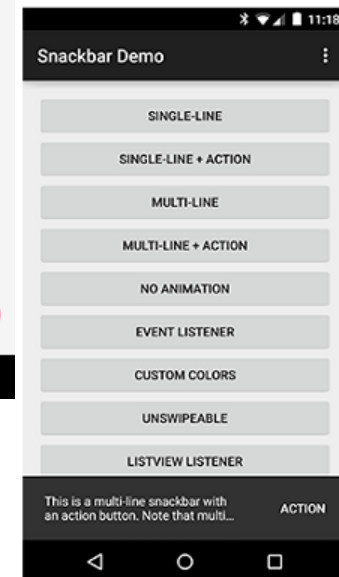
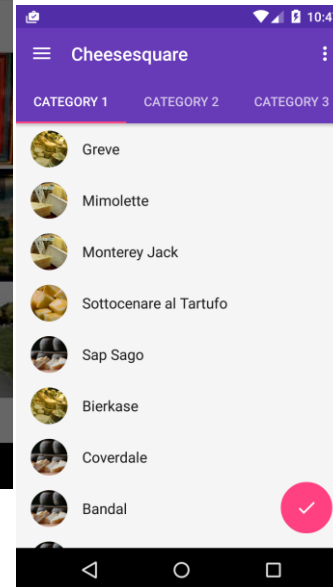
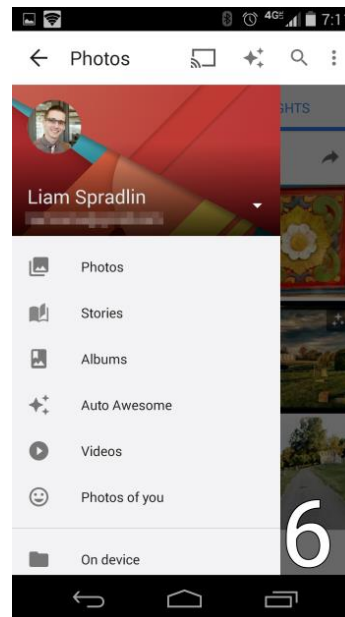




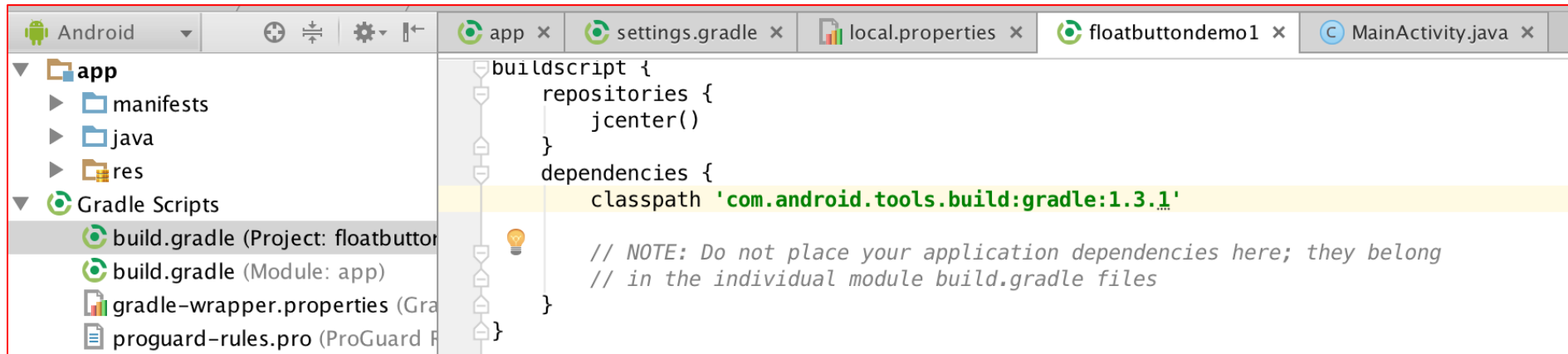
## **BÀI 7: MATERIAL DESIGN**

### **PHẦN 1: GIỚI THIỆU VÀ CÁC VIEW CƠ BẢN**

- ❑ I/O 2015 Google công bố thư viện hỗ trợ mới được gọi là Material Design. Được hỗ trợ từ android 2.3 trở lên.
- ❑ Bao gồm một số control và tính năng mới:
  - ❖ FloatingActionButton
  - ❖ TabLayout
  - ❖ NavigationView
  - ❖ SnackBar
  - ❖ TextInputLayout
  - ❖ CoordinatorLayout
  - ❖ Percent Layout
  - ❖ Vector Drawables
  - ❖ Animating view hierarchies
  - ❖ Bottom Navigation Views
  - ❖ .....



## ❑ Kiểm tra build.gradle thấp nhất 1.2.3



## ❑ Thêm support:design:



- ❑ Ngoài ra tùy theo loại control mà phải gắn thêm các thư viện. vd:

```
dependencies {  
    compile fileTree(include: ['*.jar'], dir: 'libs')  
    compile 'com.android.support:appcompat-v7:23.0.1'  
    compile 'com.android.support:design:23.0.1'  
    compile 'com.android.support:recyclerview-v7:23.0.1'  
    compile 'com.android.support:percent:23.0.1'  
}
```

- ❑ Cài thêm Android Support Repository

❑ Tương tự như Toast nhưng có thêm 1 nút để tương tác

```
Snackbar.make(findViewById(R.id.root), "chac chua", 5000)
    .setActionTextColor(Color.RED)
    .setAction("OK", new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            //xu ly gi do o day
        }
    })
    .show();
```





❑ Bộc ngoài EditText để tạo hiệu ứng cho EditText




❑ Có thể bắt sự kiện `AddTextChangedListener` và trong hàm `OnTextChanged` kiểm tra validation và thông báo lỗi

```
TextInputLayout t=(TextInputLayout)findViewById(R.id.textinputlayout);  
t.setError("nhap dum cai ten");
```

❑ Có thể chỉnh thêm màu sắc trong style

```
<item name="colorControlNormal">#d32f2f</item>  
<item name="colorControlActivated">#ff5722</item>  
<item name="colorControlHighlight">#f44336</item>
```



Slide7a

Nhap Ten

khoa

nhap dum cai ten

Nhap Tuo

- ❑ Chia layout theo tỉ lệ phần trăm
- ❑ Nhúng Percent

```
dependencies {  
    compile fileTree(dir: 'libs', include: ['*.jar'])  
    compile 'com.android.support:appcompat-v7:23.0.1'  
    compile 'com.android.support:percent:23.0.1'  
}
```

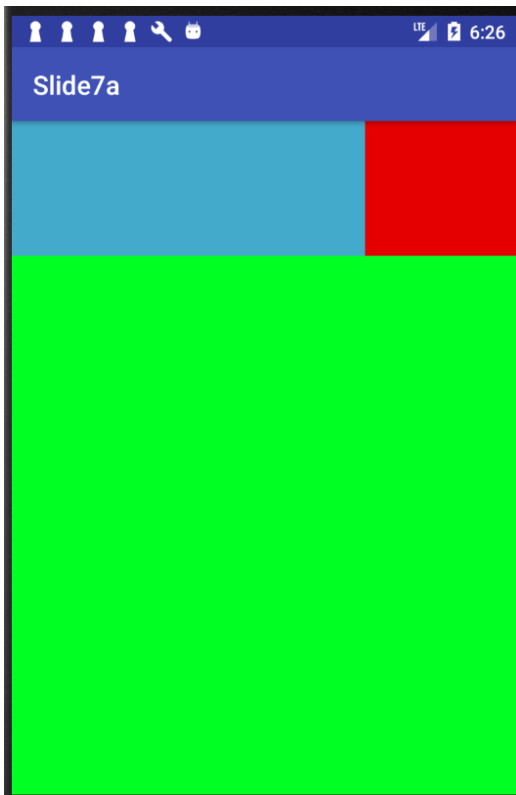
- ❑ Có 2 loại percent layout là PercentRelativeLayout và PercentFrameLayout

## □ Layout

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.percent.PercentRelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent" android:layout_height="match_parent">
    <View
        android:id="@+id/top_left"
        android:layout_width="0dp"
        android:layout_height="0dp"
        android:layout_alignParentTop="true"
        android:background="#ff44aacc"
        app:layout_heightPercent="20%"
        app:layout_widthPercent="70%" />

    <View
        android:id="@+id/top_right"
        android:layout_width="0dp"
        android:layout_height="0dp"
        android:layout_alignParentTop="true"
        android:layout_toRightOf="@+id/top_left"
        android:background="#ffe40000"
        app:layout_heightPercent="20%"
        app:layout_widthPercent="30%" />
```

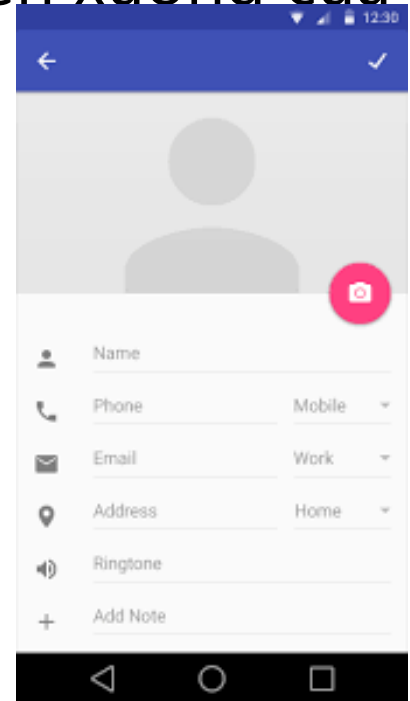
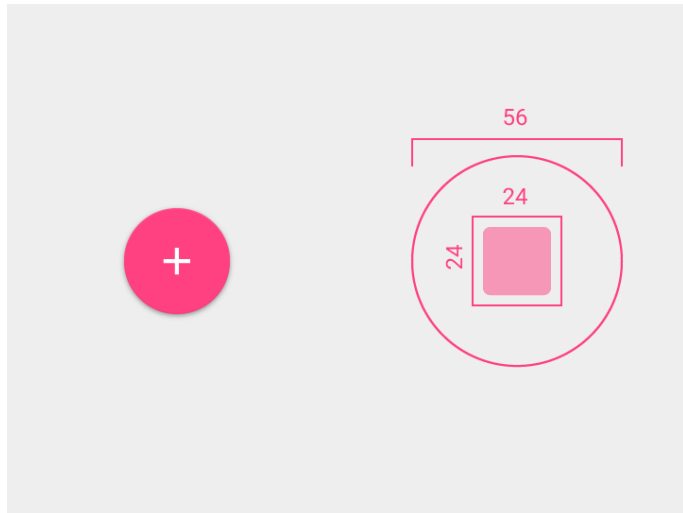
```
<View
    android:id="@+id/bottom"
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_below="@+id/top_left"
    android:background="#ff00ff22"
    app:layout_heightPercent="80%" />
</android.support.percent.PercentRelativeLayout>
```



□ Các thuộc tính có thể dùng percent

- ❖ heightPercent
- ❖ widthPercent
- ❖ marginBottomPercent
- ❖ marginEndPercent
- ❖ marginLeftPercent
- ❖ marginPercent
- ❖ marginRightPercent
- ❖ marginStartPercent

- ❑ FloatingActionButton thường đặt ở góc phải dưới của màn hình, thường hở dưới 16dp cho phone và 24dp cho tablet, kích thước của nó nên là 24dp
- ❑ Có 1 dạng vị trí nữa là nằm chồng lên góc dưới của phần Header
- ❑ hoặc có thể có animation ẩn hiện khi trượt lên xuống của một listView



## □ Layout:

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.design.widget.CoordinatorLayout
    android:id="@+id/main_content"
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <ListView
        android:id="@+id/lvToDoList"
        android:layout_width="match_parent"
        android:layout_height="match_parent">
    </ListView>
    <android.support.design.widget.FloatingActionButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom|right"
        android:layout_margin="16dp"
        android:src="@mipmap/ic_launcher_round"
        app:layout_anchor="@id/lvToDoList"
        app:layout_anchorGravity="bottom|right|end" />
</android.support.design.widget.CoordinatorLayout>
```

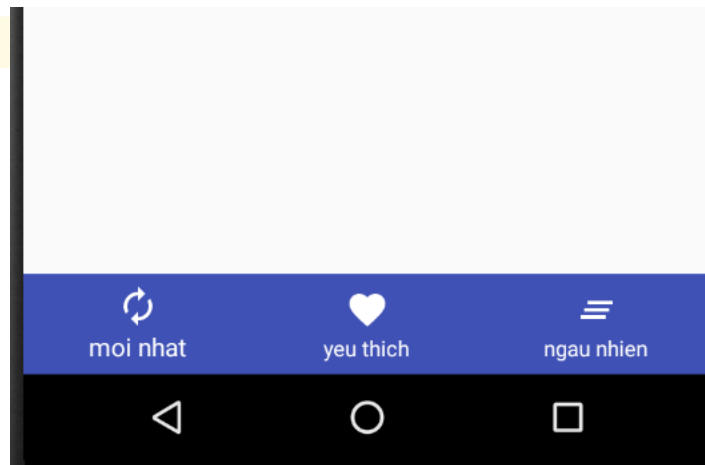
- ❑ Menu nằm bên dưới của Activity
- ❑ Thực hiện:
  - ❖ Tạo file menu (giống cách tạo context/option menu)
  - ❖ Thêm view BottomNavigationView

```
<android.support.design.widget.BottomNavigationView
    android:id="@+id/bottom_navigation"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    app:itemBackground="@color/colorPrimary"
    app:itemIconTint="#ffffff"
    app:itemTextColor="#ffffff"
    app:menu="@menu/menu_bottom" />
```



## ❖ Bắt sự kiện

```
bnv.setOnItemClickListener(  
    new BottomNavigationView.OnNavigationItemSelectedListener() {  
        @Override  
        public boolean onNavigationItemSelected(MenuItem item) {  
            if(item.getItemId()==R.id.action_moi)  
                Toast.makeText(Main4Activity.this, "moi", Toast.LENGTH_SHORT).show();  
            if(item.getItemId()==R.id.action_yeu_thich)  
                Toast.makeText(Main4Activity.this, "yeu", Toast.LENGTH_SHORT).show();  
            if (item.getItemId()==R.id.action_ngau_nhien)  
                Toast.makeText(Main4Activity.this, "ngau nhien", Toast.LENGTH_SHORT).show();  
            return false;  
        }  
    });
```





SnackBar **DEMO**

TextInputLayout

PercentLayout

FloatActionButton

BottomNavigationView





## **BÀI 7: MATERIAL DESIGN**

### **PHẦN 2: CÁC VIEW TẠO HIỆU ỨNG**

❖ CoordinatorLayout: là một nâng cấp mạnh mẽ của FrameLayout. Mục đích chính của nó là làm top-layout cho các thiết kế phức tạp, bên trong nó có thể có các view con tương tác với nhau (nhận thao tác trực tiếp để phóng to thu nhỏ bar, ẩn hiện ActionBar....)

- ❖ AppBarLayout: là một linearLayout dạng vertical thực hiện các tính năng mới của material design, cụ thể là hỗ trợ tính năng hiệu ứng khi cuộn cho các view con của nó.
- ❖ Nó nên dùng chung với CoordinatorLayout nếu không hầu hết chức năng của nó sẽ không hoạt động.
- ❖ Nó cũng cần một view anh em (view ngang cấp) bên dưới dùng để cuộn, vd như 1 NestedScrollView và dùng ScrollingViewBehavior để liên kết với nhau. Behavior này có sẵn 1 chuỗi tài nguyên chứa nó:  
**@string/appbar\_scrolling\_view\_behavior**

- ❖ Nó là con trực tiếp của AppBarLayout và bao bọc (là cha) của ToolBar. Nó thường chứa:
  - Collapsing title: (tiêu đề rút gọn), tiêu đề lớn nhỏ khi cuộn.
  - Content scrim: màu hoặc hình của vùng content ẩn hiện khi cuộn
  - Status bar scrim: màu hoặc hình của statusbar ẩn hiện khi cuộn
  - Parallax scrolling children: View sẽ ẩn hiện khi ta cuộn
  - Pinned position children: View được ghim cố định (không bị ẩn, chỉ di chuyển) khi ta cuộn.

- ❑ Phối hợp CoordinatorLayout, AppBarLayout, CollapsingToolBarLayout,... để tạo hiệu ứng trượt AppBarLayout và FloatingActionButton



```
<android.support.design.widget.CoordinatorLayout ...>
```

```
<android.support.design.widget.AppBarLayout
```

```
    android:id="@+id/main.appbar" ....>
```

```
    <android.support.design.widget.CollapsingToolbarLayout
```

```
        app:layout_scrollFlags="scroll|exitUntilCollapsed"
```

```
        app:contentScrim="?attr/colorPrimary" ....>
```

```
        <ImageView
```

```
            app:layout_collapseMode="parallax" .... />
```

```
        <android.support.v7.widget.Toolbar
```

```
            app:layout_collapseMode="pin" ...../>
```

```
    </android.support.design.widget.CollapsingToolbarLayout>
```

```
</android.support.design.widget.AppBarLayout>
```

```
<android.support.v4.widget.NestedScrollView
```

```
    app:layout_behavior="@string/appbar_scrolling_view_behavior" ...>
```

```
    <TextView ..... />
```

```
</android.support.v4.widget.NestedScrollView>
```

```
<android.support.design.widget.FloatingActionButton
```

```
    app:layout_anchor="@id/main.appbar"
```

```
    app:layout_anchorGravity="bottom|right|end" ...../>
```

```
</android.support.design.widget.CoordinatorLayout>
```





# DEMO

Phối hợp các view  
để tạo hiệu ứng ẩn  
hiện khi trược



- ☐ Material Design
- ☐ Một số view thông dụng
- ☐ Các view phối hợp tạo hiệu ứng





**Cảm ơn**