# Android Tab View Application

In this tutorial we will create three separate activities for three tab screens. This is similar to the tabbed view application in iOS.

**Step 1**. Create a new project **File -> New -> Android Project** and give activity name like **Tabs**.

**Step 2**. Open your main activity class and extend the class from *TabActivity* as follows: (ignore any depreciation warnings)

```
public class MainActivity extends TabActivity {
```

**Step 3**. Now open your **main.xml** under **res -> layout** folder and type the following code to create our desired UI look:

```
<?xml version="1.0" encoding="utf-8"?>
<TabHost
xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@android:id/tabhost"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">
    <LinearLayout
        android:orientation="vertical"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent">
        <TabWidget
            android:id="@android:id/tabs"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content" />
        <FrameLayout
            android:id="@android:id/tabcontent"
            android:layout_width="fill_parent"
            android:layout_height="fill_parent"/>
    </LinearLayout>
</TabHost>
```

**Step 4**. Now we need 3 activities and 3 xml layouts for three tabs. So create three activities by right click on your package folder and create classes and name them as ***PhotosActivity.java***, ***SongsActivity.java*** and ***VideosActivity****.java*. Type the following code respectively.

**Right Click on package folder -> New -> Class**

» PhotosActivity.java

```
import android.app.Activity;
import android.os.Bundle;

public class PhotosActivity extends Activity {
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.photos_layout);
    }
}
```

» SongsActivity.java

```
import android.app.Activity;
import android.os.Bundle;

public class SongsActivity extends Activity {
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.songs_layout);
    }
}
```

» VideosActivity.java

```
import android.app.Activity;
import android.os.Bundle;

public class VideosActivity extends Activity {
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.videos_layout);
    }
}
```

**Step 5**. Now create 3 xml layouts by **right clicking on res/layout -> New -> Android XML** File and name them as ***photos_layout.xml***,

***songs_layout.xml*** and ***videos_layout.xml*** and type the following code in respective files.

» photos_layout.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">

  <!-- Screen Design for Photos -->
  <TextView android:text="PHOTOS HERE"
          android:padding="15dip"
          android:textSize="18dip"
          android:layout_width="fill_parent"
          android:layout_height="wrap_content"/>

</LinearLayout>
```

» songs_layout.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">

  <!-- Screen Design for the SONGS -->
  <TextView android:text="SONGS HERE"
          android:padding="15dip"
          android:textSize="18dip"
          android:layout_width="fill_parent"
          android:layout_height="wrap_content"/>
</LinearLayout>
```

» videos_layout.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">

  <!--  Screen Design for VIDEOS -->
  <TextView android:text="VIDEOS HERE"
          android:padding="15dip"
          android:textSize="18dip"
          android:layout_width="fill_parent"
          android:layout_height="wrap_content"/>
</LinearLayout>
```

**Step 6**. Each and every tab needs an icon so design icons for each tab. We need three dimensions of each icon. Design each icon in *48 x 48 px*, *32 x 32 px* and *24 x 24 px* and place them in *drawable-hdpi*, *drawable-mdpi* and *drawable-ldpi* respectively. I will just stick generic sized icons in the **drawable** folder.

**Step 7**. Android icon states will be define in xml files with default and hover state configurations. For three icons we need the icon state configuration files.

Create three 3 xml files under the drawable directory. Type the following code for icon states.

» icon_photos_tab.xml

```
<?xml version="1.0" encoding="utf-8"?>
<selector
xmlns:android="http://schemas.android.com/apk/res/android">
    <!-- When selected, use grey -->
    <item android:drawable="@drawable/photos_gray"
         android:state_selected="true" />
    <!-- When not selected, use white-->
    <item android:drawable="@drawable/photos_white" />
</selector>
```

» icon_songs_tab.xml

```
<?xml version="1.0" encoding="utf-8"?>
<selector
```

```
xmlns:android="http://schemas.android.com/apk/res/android">
    <!-- When selected, use grey -->
    <item android:drawable="@drawable/songs_gray"
          android:state_selected="true" />
    <!-- When not selected, use white-->
    <item android:drawable="@drawable/songs_white" />
</selector>
```

» icon_videos_tab.xml

```
<?xml version="1.0" encoding="utf-8"?>
<selector
xmlns:android="http://schemas.android.com/apk/res/android">
    <!-- When selected, use grey -->
    <item android:drawable="@drawable/videos_gray"
          android:state_selected="true" />
    <!-- When not selected, use white-->
    <item android:drawable="@drawable/videos_white" />
</selector>
```

**Step 8**. Open **MainActivity.java** and type the following code.

In the following code we are creating three *TabSepcs* and adding them to *TabHost*.

» AndroidTabLayoutActivity.java

```
import android.app.TabActivity;
import android.content.Intent;
import android.os.Bundle;
import android.widget.TabHost;
import android.widget.TabHost.TabSpec;

public class AndroidTabLayoutActivity extends TabActivity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
```

```
        TabHost tabHost = getTabHost();

        // Tab for Photos
        TabSpec photospec = tabHost.newTabSpec("Photos");
        // setting Title and Icon for the Tab
        photospec.setIndicator("Photos",
getResources().getDrawable(R.drawable.icon_photos_tab));
        Intent photosIntent = new Intent(this, PhotosActivity.class);
        photospec.setContent(photosIntent);

        // Tab for Songs
        TabSpec songspec = tabHost.newTabSpec("Songs");
        songspec.setIndicator("Songs",
getResources().getDrawable(R.drawable.icon_songs_tab));
        Intent songsIntent = new Intent(this, SongsActivity.class);
        songspec.setContent(songsIntent);

        // Tab for Videos
        TabSpec videospec = tabHost.newTabSpec("Videos");
        videospec.setIndicator("Videos",
getResources().getDrawable(R.drawable.icon_videos_tab));
        Intent videosIntent = new Intent(this, VideosActivity.class);
        videospec.setContent(videosIntent);

        // Adding all TabSpec to TabHost
        tabHost.addTab(photospec); // Adding photos tab
        tabHost.addTab(songspec); // Adding songs tab
        tabHost.addTab(videospec); // Adding videos tab
    }
}
```

**Step 9**. Now everything is ready and before running your project make sure that you an entry of new activity name in **AndroidManifest.xml** file.

Open your **AndroidManifest.xml** file and modify the code as below

» AndroidManifest.xml

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

```
    android:versionCode="1"
    android:versionName="1.0">
  <uses-sdk android:minSdkVersion="8" />

  <application android:icon="@drawable/icon"
android:label="@string/app_name">
      <activity android:name=".MainActivity"
            android:label="@string/app_name">
        <intent-filter>
          <action android:name="android.intent.action.MAIN" />
          <category
android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
      </activity>

      <!--  Songs Activity -->
      <activity android:name=".SongsActivity" />

      <!--  Videos Activity -->
      <activity android:name=".VideosActivity" />

      <!--  Photos Activity -->
      <activity android:name=".PhotosActivity" />

  </application>
</manifest>
```

**Step 10**. Run your project by **right clicking on your project folder -> Run As -> 1 Android Application**. You will see simple tabs app with navigation!