Inducesmile
Mobile developer community for beginners and students

ANDROID	KOTLIN	IOS (SWIFT)	HYBRID APP ▾	MOBILE GAMES	IOT – VR
TIPS	CODING HELP	BUY CODE			

🏠 **Home** ▸ **Android** ▸ Android Bottom Navigation View Example

ANDROID BOTTOM NAVIGATION VIEW EXAMPLE

In this tutorial, we are going to learn how to create android bottom navigation view using the design support library in android.

Android Bottom Navigation View is another way to navigate through related pages in your android application. It is suited to navigate between 3 to 5 pages. If you are planning to add more than 5 different Activity or Fragment pages in your top-level destination, it is highly recommended to use Android Navigation Drawer.

You can as well use Android Tab for 1 or 2 top-level destination. In order to design efficient android Bottom Navigation View that follows android material design principles, I will suggest you read **Google User Guide for Bottom Navigation View in android.**

Before the new version of the android design support library V25, there are many issues that developers face when working with android

ANDROID SOURCE CODE POLL

Which of the below listed source codes should we publish next month?

Multi Restaurant Food Ordering App (27%, 39 Votes)

Stock and Inventory App (18%, 26 Votes)

Android Dating App (16%, 23

If you are interested to learn how to implement **android Tab** and **Navigation drawer**, you can read my tutorials on these topics.
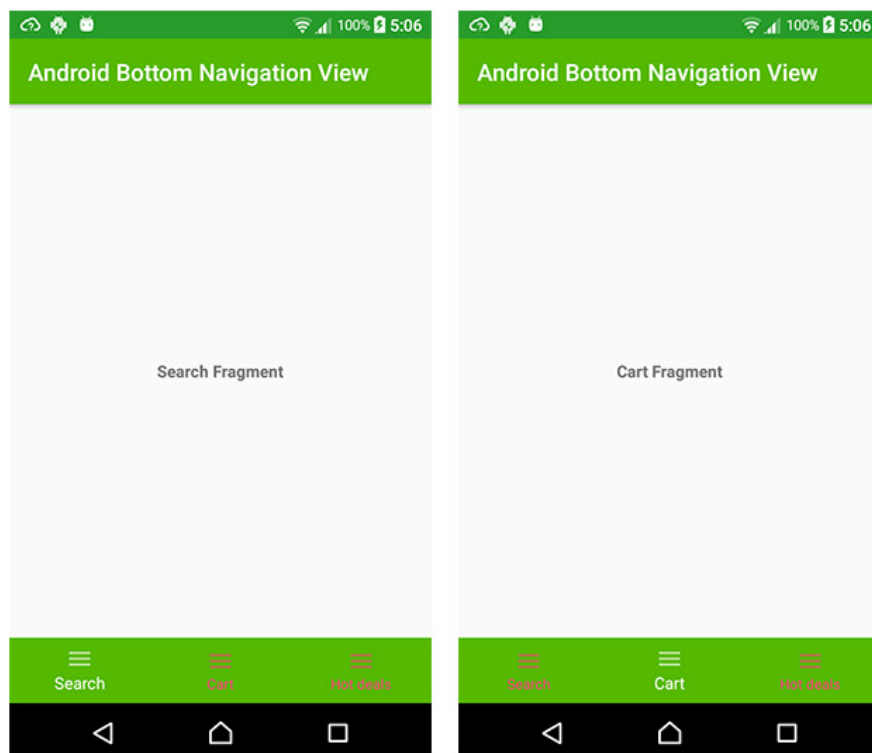
So we are going to use the Android Support Library v25 in this tutorial to implement android Bottom Navigation View.

In other to get access to Android Bottom Navigation View, we are going to update our project app   build.gradle   file.

Also, we will make use of 3 Fragment classes that will be associated to each menu.

In other to get a visual understanding of what we are going to create in this android tutorial, I have add below some screen shots from the application.

SOME SCREENSHOT FROM THE APPLICATION

LIKE US ON FACEBOOK

RECENT POSTS

- Camera Translator All Language App
- Why MVVM and How to execute MVVM combined with Data Binding and LiveData in Four Simple Steps
- Camera Translator All Languages – Android Project idea source code for students
- Note Scan Android App -Project idea source code for students
- How To Build Android WhatsApp Clone Chat Application
- Android Car Rental App with Web Admin Panel – Project Idea Source Code For Students

Accept

Lets start to soil our hands in code. Start up your IDE. For this tutorial, I am using the following tools and environment, feel free to use what works for you.

Windows 10

Android Studio

Sony Xperia ZL

Min SDK 14

Target SDK 24

To create a new android application project, follow the steps as stipulated below.

Go to File menu

Click on New menu

Click on Android Application

Enter Project name: AndroidBottomNavigationView

Package: com.inducesmile.androidbottomnavigationview

Select Empty Activity

Name your activity: MainActivity

Keep other default selections

Continue to click on next button until Finish button is active, then click on Finish Button.

Now that we have created our android project, if you are not using *targetSdkVersion 25,* you can update your android SDK version to 25.

- Android Retrofit 2 with JSON API Example
- Android Room with LiveData Example Tutorial
- Android UI App for Food Ordering and Delivery from Multiple Restaurants

Accept

```
compile 'com.android.support:appcompat-v7:25.0.0'
```

```
compile 'com.android.support:appcompat-v7:25.0.0'
compile 'com.android.support:design:25.0.0'
```

Now the complete code in the page will look like below.

```
apply plugin: 'com.android.application'
android {
    compileSdkVersion 25
    buildToolsVersion "24.0.1"
    defaultConfig {
        applicationId "com.inducesmile.androidbottomnavi
        minSdkVersion 14
        targetSdkVersion 25
        versionCode 1
        versionName "1.0"
        testInstrumentationRunner "android.support.test.
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('progua
        }
    }
}
dependencies {
    compile fileTree(dir: 'libs', include: ['*.jar'])
    androidTestCompile('com.android.support.test.espress
        exclude module: 'support-annotations'
    })
    compile 'com.android.support:appcompat-v7:25.0.0'
    compile 'com.android.support:design:25.0.0'
    compile 'com.android.support:support-v4:25.0.0'
    testCompile 'junit:junit:4.12'
}
```

Accept

We are going to update our project strings.xml file located in the values
folder inside the res folder. Open the file and add the code below to it.

```xml
<resources>
    <string name="app_name">Android Bottom Navigation Vi
    <string name="menu">Menu</string>
    <string name="action_search">Search</string>
    <string name="action_cart">Cart</string>
    <string name="action_hot_deals">Hot deals</string>
    <!-- TODO: Remove or change this placeholder text --
    <string name="hello_blank_fragment">Hello blank frag
    <string name="search_fragment">Search Fragment</stri
    <string name="cart_fragment">Cart Fragment</string>
    <string name="hot_deals_fragment">Hot Deals Fragment
</resources>
```

COLORS.XML

Open the colors.xml file in the same location as the strings.xml file and
add the code below to the file.

```xml
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#56B900</color>
    <color name="colorPrimaryDark">#289C2B</color>
    <color name="colorAccent">#FF4081</color>
    <color name="colorPrimaryLight">#84FF19</color>
    <color name="colorWhite">#ffffff</color>
    <color name="colorBlack">#000000</color>
    <color name="colorDivider">#2FE897</color>
    <color name="colorBorder">#a7a4a4</color>
    <color name="colorExtra">#fffac1</color>
</resources>
```

CREATE ACTIVITY CLASS

Accept

Lets open that activity_main.xml layout file located at the res/layout folder. We are going to add BottomNavigationView widget to the layout View group. In order to display a different Fragment depending on which menu of the Bottom Navigation View is clicked, we will add a Framelayout that will hold the Fragment class.

Add the code below to the layout file.

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.co
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.inducesmile.androidbottomnavigati
    <FrameLayout
        android:id="@+id/main_container"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_above="@+id/bottom_navigation"
        android:layout_alignParentTop="true">
    </FrameLayout>
    <android.support.design.widget.BottomNavigationView
        android:id="@+id/bottom_navigation"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        app:itemBackground="@color/colorPrimary"
        app:itemIconTint="@color/color_state_list"
        app:itemTextColor="@color/color_state_list"
        android:layout_alignParentBottom="true">
    </android.support.design.widget.BottomNavigationView
</RelativeLayout>
```

MAINACTIVITY CLASS

The MainActivity class will use the activity_main.xml layout file we

Accept

To get the instance of our BottomNavigationView, we will use the findViewById() method of the Activity class.

The create instance will call the inflateMenu() method of its class to inflate a menu it will use as its interface. We are going to create the menu UI soon.

The next step is to wire an event listener to the BottonNavigationView that will monitor when each menu item is selected.

Finally, will create an instance of a Fragment class depending of which menu is selected and replace the FrameLayout with the Fragment instance.

Open the MainActivity.java class and add the code below to it.

```
import android.os.Bundle;                                          </>
import android.support.annotation.NonNull;
import android.support.design.widget.BottomNavigationVie
import android.support.v4.app.Fragment;
import android.support.v4.app.FragmentManager;
import android.support.v4.app.FragmentTransaction;
import android.support.v7.app.AppCompatActivity;
import android.view.MenuItem;
public class MainActivity extends AppCompatActivity {
    private static final String TAG = MainActivity.class
    private BottomNavigationView bottomNavigation;
    private Fragment fragment;
    private FragmentManager fragmentManager;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        bottomNavigation = (BottomNavigationView)findVie
        bottomNavigation.inflateMenu(R.menu.bottom_menu)
        fragmentManager = getSupportFragmentManager();
        bottomNavigation.setOnNavigationItemSelectedList
            @Override
            public boolean onNavigationItemSelected(@Non
                int id = item.getItemId();
```

```
                fragment = new CartFragment();
                break;
            case R.id.action_hot_deals:
                fragment = new DealsFragment();
                break;
        }
        final FragmentTransaction transaction =
        transaction.replace(R.id.main_container,
        return true;
    }
 });
}
}
```

CREATE MENU FOR BOTTOM NAVIGATION VIEW

BOTTOM_MENU.XML

We are going to create a menu resource layout file. As stated in the MainActivity class, the instance of the BottomNavigationView inflates a menu layout.

Under the res folder of your project, create a menu folder if it does not exist. Inside the menu folder, create a menu resource file and name it bottom_menu.xml.

Open the file and add the code below.

```
                                                    </>
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:tools="http://schemas.android.com/tools"
    xmlns:android="http://schemas.android.com/apk/res/an
    xmlns:app="http://schemas.android.com/apk/res-auto">
    <item
        android:id="@+id/action_search"
        android:title="@string/action_search"
        app:showAsAction="ifRoom"
```

Accept

```
        android:id="@+id/action_cart"
        android:title="@string/action_cart"
        app:showAsAction="ifRoom"
        android:enabled="true"
        android:icon="@drawable/ic_action_menu">
    </item>
    <item
        android:id="@+id/action_hot_deals"
        android:title="@string/action_hot_deals"
        app:showAsAction="ifRoom"
        android:enabled="true"
        android:icon="@drawable/ic_action_menu">
    </item>

</menu>
```

CREATE FRAGMENT CLASSES

We are going to create three different Fragment classes
DealsFragment.java, CartFragment.java and SearchFragment.java.

Each of this Fragments will display their names when they are selected
by click a BottomNavigationView menu item.

Their implementation is very simple but feel free to extend it according
to your project requirements.

DEALSFRAGMENT.JAVA

```
                                                        </>
import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
public class DealsFragment extends Fragment {
```

Accept

```
        }
    }

<FrameLayout xmlns:android="http://schemas.android.com/a
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.inducesmile.androidbottomnavigati
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/hot_deals_fragment"
        android:textStyle="bold"
        android:layout_gravity="center"/>
</FrameLayout>
```

CARTFRAGMENT.JAVA

```
import android.os.Bundle;                                          </>
import android.support.v4.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
public class CartFragment extends Fragment {
    public CartFragment() {
    }
    @Override
    public View onCreateView(LayoutInflater inflater, Vi
        return inflater.inflate(R.layout.fragment_cart,
    }
}


<FrameLayout xmlns:android="http://schemas.android.com/a
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.inducesmile.androidbottomnavigati
```

Accept

```xml
        android:textStyle="bold"
        android:layout_gravity="center"/>
</FrameLayout>
```

SEARCHFRAGMENT.JAVA

```java
import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
public class SearchFragment extends Fragment {
    public SearchFragment() {
    }
    @Override
    public View onCreateView(LayoutInflater inflater, Vi
        View view = inflater.inflate(R.layout.fragment_s
        return view;
    }
}

<FrameLayout xmlns:android="http://schemas.android.com/a
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.inducesmile.androidbottomnavigati
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:textStyle="bold"
        android:text="@string/search_fragment" />
</FrameLayout>
```

This brings us to the end of this tutorial. I hope that you have learn something. Run your app and see for yourself.

Accept

Remember to subscribe with your email address to be among the first
to receive my new android blog post once it is published.

OTHER INTERESTING POSTS:

How to play YouTube video
inside Android WebView
using Video Url

Android Studio – Add .Jar
File as Android Library

Android Json Parsing using
Gson and RecyclerView

Android Content Provider -
Access SQLite database in
one application from
another application

Android Online Quiz With
Web Based Admin Panel –
Project Idea Source Code
for Students

How To Build Android
WhatsApp Clone Chat
Application

Tags: Android Bottom Navigation, Android Bottom Navigation View,

Accept

Inducesmile

I learn and write about Android, iOS, Javascript, Php, Node.js, React Native, Mobile Game, Virtual Reality and Internet of Things

5 COMMENTS

Django | March 22, 2017   Log in to Reply

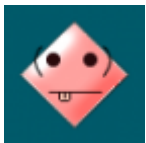Awesome tutorial bro!! u rock…plz keep on posting tutorials like these 🙂

Henry Author | March 23, 2017

Log in to Reply

I appreciate your kind words. Thanks so much.

Regards
Henry

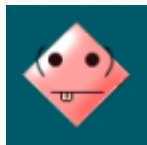**aldis** | May 27, 2017   Log in to Reply

Hi Henry, how to remove blank fragment in this tutorial? it is really disturbing me
i'm sorry my english bad 😁

Accept

I don't understand what you are referring to as blank Fragment? Thanks

aldis | May 29, 2017   Log in to Reply

and, can you send me a link code of this tutorial to aldyputraw@gmail.com ?
thanks

ADD A COMMENT

You must be logged in to post a comment.

INDUCESMILE – ANDROID TUTORIAL, ANDROID APPS, ANDROID STUDIO, ANDROID SDK, ANDROID DEVELOPMENT COPYRIGHT © 2018.

PROGRAMMING HELP   ANDROID TUTORIALS   CONTACT US   PRIVACY POLICY

Accept