Android Black Book

Easy to Learn Android

Friday, 5 February 2016

Creating a System Overlay (Always on Top over all Apps) in Android

In this post, we have learn about creating a system overlay (Always on Top over all Apps).

1)Add the following permission to Android Manifest.XML.

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

2)Add WindowManager.

The Android WindowManager is a system service, which is responsible for managing the z-ordered list of windows, which windows are visible, and how they are laid out on screen. Among other things, it automatically performs window transitions and animations when opening or closing an app or rotating the screen.

Every activity has a Window that is used to display its content on the screen. When you call setContentView on an activity, it attaches that view to the activity's default window. The default window fills the screen, so that your activity's window hides any other activities -- the WindowManager will display whichever window is on top. So normally you don't need to worry about windows - you just create an activity and Android will do the rest for you.

But you need to interact with the WindowManager if you want to do something unusual like create floating windows that don't fill the screen. If you want to create a floating window that is visible in front of other applications, you can't use an activity because your activity will stop when another app comes to the foreground, and its window will be hidden or destroyed. Instead you need to display a window from a background service.

For example:

```
 private WindowManager windowManager;
windowManager = (WindowManager) getSystemService(WINDOW_SERVICE);
 LayoutInflater inflater = (LayoutInflater)getSystemService(LAYOUT_INFLATER_SERVICE);
        WindowManager.LayoutParams params = new WindowManager.LayoutParams(
                WindowManager.LayoutParams.WRAP_CONTENT,
                WindowManager.LayoutParams.WRAP_CONTENT,
                WindowManager.LayoutParams.TYPE_PHONE,
                WindowManager.LayoutParams.FLAG_NOT_FOCUSABLE,
                PixelFormat.TRANSLUCENT);
       params.gravity = Gravity.TOP | Gravity.LEFT;
        params.x = 0;
        params.y = 100;
        chatheadView = (RelativeLayout) inflater.inflate(R.layout.activity_alert_dialog, null);
windowManager.addView(chatheadView, params);
```

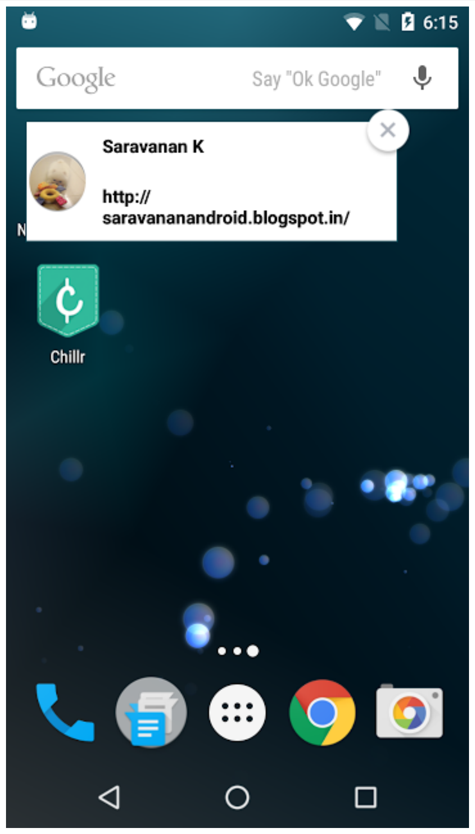
For Lollipop and Marshmallow devices, you need to add following code for Notification Overlay:

```
 public static void requestSystemAlertPermission(Activity context, int requestCode) {
        if (Build.VERSION.SDK_INT < Build.VERSION_CODES.M)
            return;
        final String packageName = context == null ? context.getPackageName() :
context.getPackageName();
      final Intent intent = new Intent(Settings.ACTION_MANAGE_OVERLAY_PERMISSION,
Uri.parse("package:" + packageName));
        if (context != null)
            context.startActivityForResult(intent, requestCode);
        else
```

```
    }
    @TargetApi(23)
    public static boolean isSystemAlertPermissionGranted(Context context) {
        final boolean result = Build.VERSION.SDK_INT < Build.VERSION_CODES.M ||
Settings.canDrawOverlays(context);
        return result;
    }
```

To get full code: Click Here

Result:



at February 05, 2016

4 comments:

Pradip Tambe 23 March 2016 at 06:01

awesome work bro

Reply

Anonymous 24 July 2016 at 09:34

Thank you bro

Reply

Anonymous 11 November 2016 at 16:16

Don't you think this

if (context != null)

context.startActivityForResult(intent, requestCode);

and

final String packageName = context == null ? context.getPackageName() : context.getPackageName();

is a bit strange? ;D
But except from this, your code works great. TY

Reply

Anonymous 8 June 2017 at 20:51

Any suggestions on how I could precisely place a big blue dot on my screen while I'm using an app? I need a reference point each time and I keep getting close but not bang on. I have a samsung S7.

Thanks in advance

Reply

Enter your comment...

Comment as: 小草 (Google) **Sign out**

Publish **Preview** □ Notify me

Links to this post

Create a Link

Newer Post Home Older Post

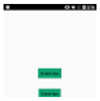
Subscribe to: Post Comments (Atom)

Blog Archive

► 2017 (3)

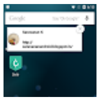
▼ 2016 (15)

- ► April (3)
- ► March (4)
- ▼ February (2)
  - Creating a System Overlay (Always on Top over all ...
  - Enable Location Services and GPS programmatically ...
- ► January (2)
- ► 2015 (19)
- ► 2014 (7)
- ► 2013 (13)



Enable Location Services and GPS programmatically (without navigating to the location settings)

In this post, we have learn about enabling Location Services and GPS programmatically (without navigating to the location settings). Fir...



Creating a System Overlay (Always on Top over all Apps) in Android

In this post, we have learn about creating a system overlay (Always on Top over all Apps). 1)Add the following permission to Android Mani...

Android Geofencing with Google Maps

 Geo-fencing is a feature in Google Play Services that allows for checking when a user has entered or left a circular area. This feature is...

Labels

- Android Interview questions

About Me

Saravanan K

G+ Follow 61

View my complete profile

Google+ Followers

Saravanan K

Add to circles

61 have me in circles View all

Translate

选择语言 ▼