# 1.修改状态栏颜色，此时状态栏依然存在

关于沉浸式状态栏相信大家都不陌生，IOS系统很早就有，android5.0及以后版本都支持给状态栏着色，而目前android主流版本还是4.4，网上通用实现4.4（API19）沉浸式状态栏也都是依赖于可以将状态栏变为透明的属性，再为其着色，主要实现代码：

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_match\_actionbar);

//只对api19以上版本有效

if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.KITKAT) {

setTranslucentStatus(true);

}

//为状态栏着色

SystemBarTintManager tintManager = new SystemBarTintManager(this);

tintManager.setStatusBarTintEnabled(true);

tintManager.setStatusBarTintResource(R.color.statusbar\_bg);

}

@TargetApi(19)

private void setTranslucentStatus(boolean on) {

Window win = getWindow();

WindowManager.LayoutParams winParams = win.getAttributes();

final int bits = WindowManager.LayoutParams.FLAG\_TRANSLUCENT\_STATUS;

if (on) {

winParams.flags |= bits;

} else {

winParams.flags &= ~bits;

}

win.setAttributes(winParams);

}

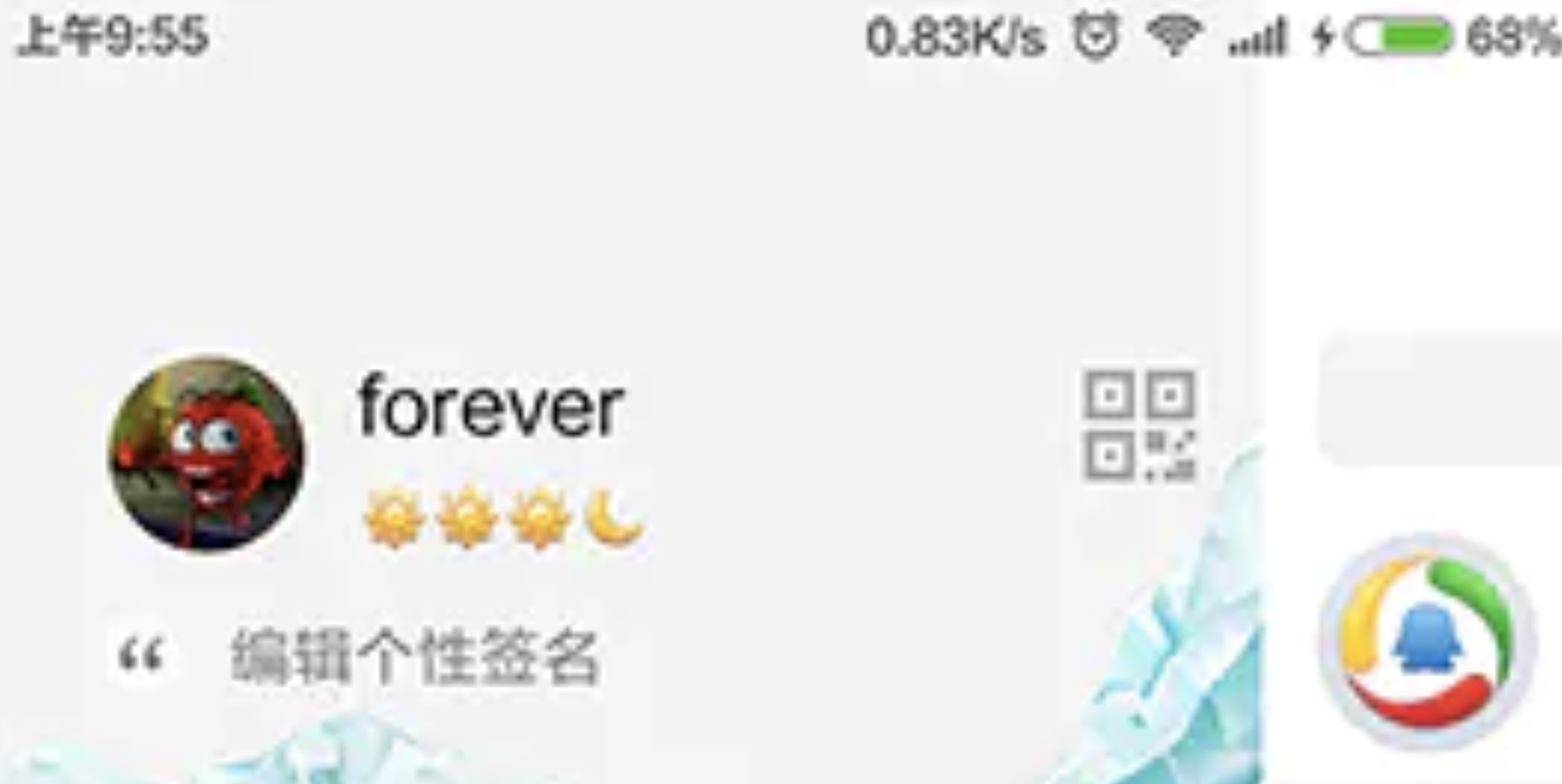
再在根布局添加以下两个属性：

android:fitsSystemWindows="true"

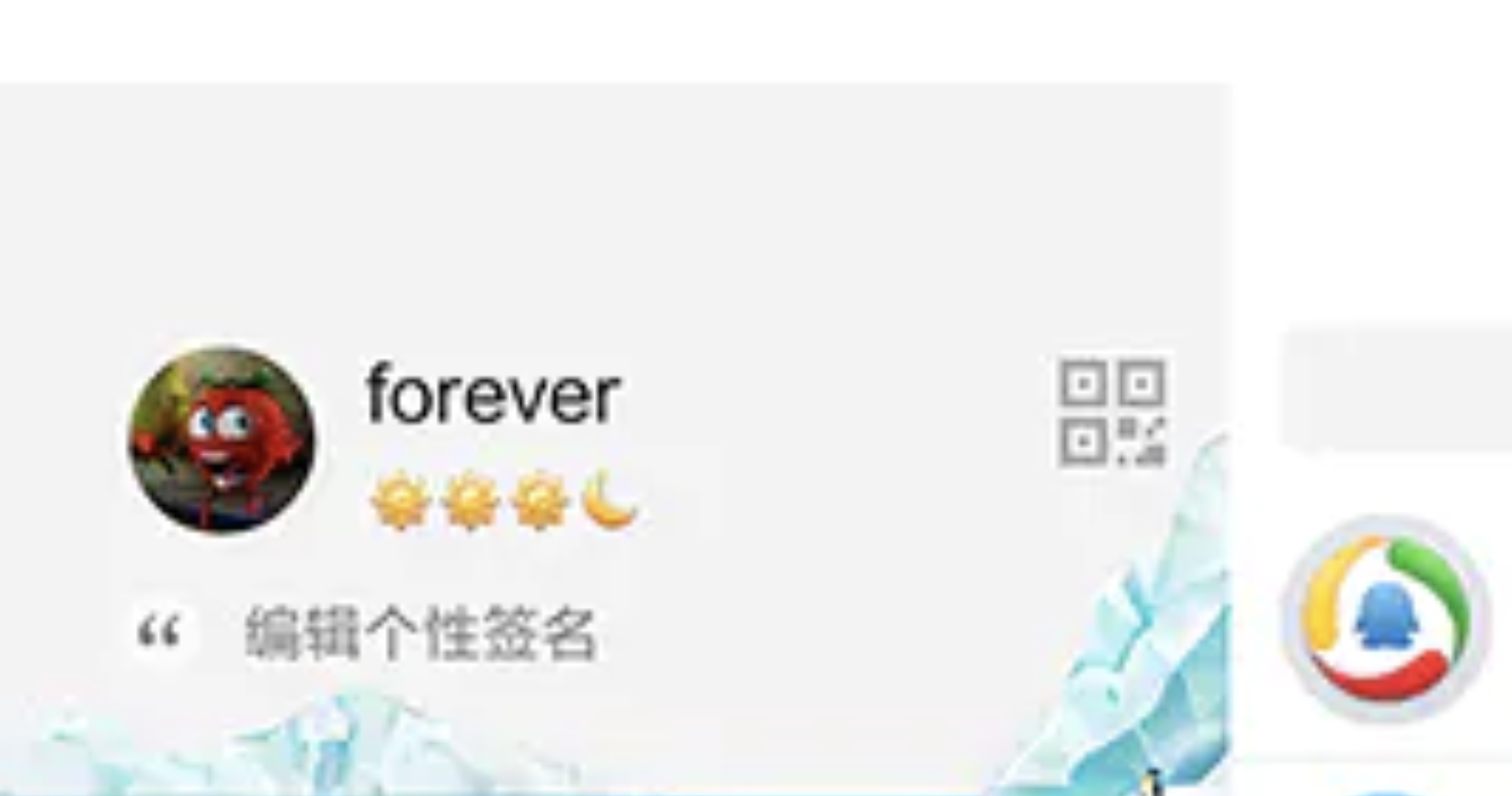
android:clipToPadding="false"

这样就可以了，以上着色使用了[SystemBarTint](https://link.jianshu.com?t=https:/github.com/jgilfelt/SystemBarTint" \t "_blank)。

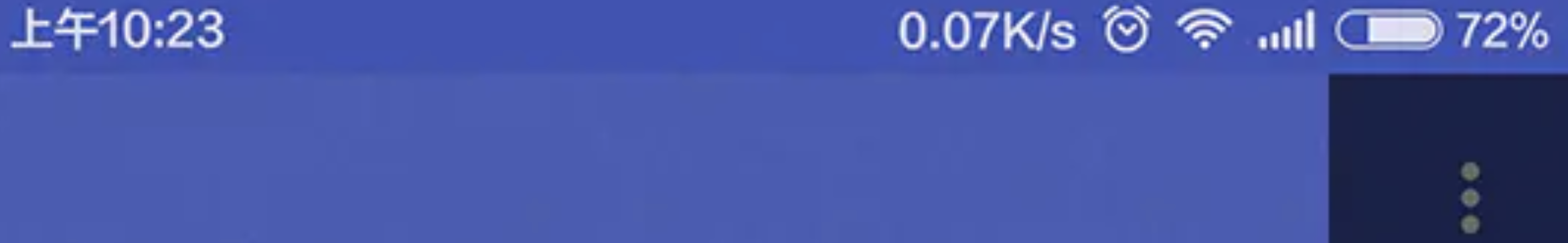
面对大多数的界面自然是没有多大问题，但是针对类似QQ这种侧滑的界面，如图：



我的手机系统版本是4.4的，如果想做成QQ侧滑背景这样的效果，使用上面的方案就变成了这样

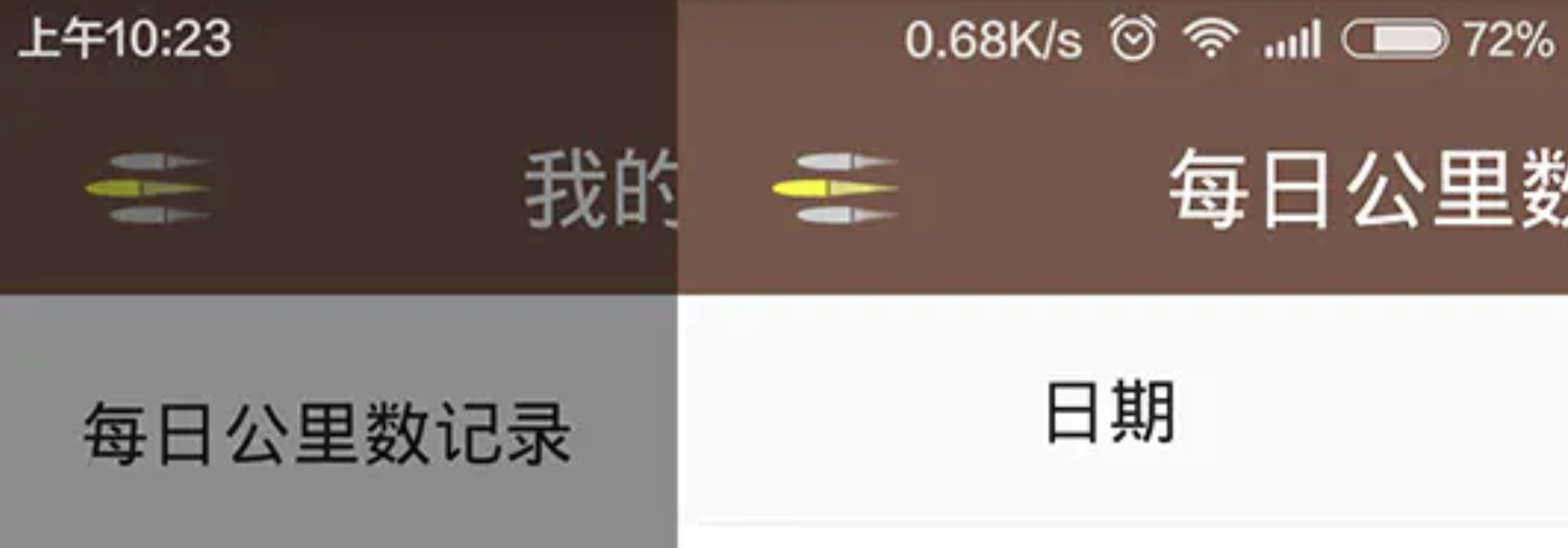


这样出来的效果就会很丑，于是才有了改进版的方案，不知QQ是否是这样做的。  
除了上述的缺陷以外，还有一点看着不是很舒服，就是当我使用抽屉菜单或者滑动返回效果的时候是这样的



状态栏并没有阴影效果

我想要的效果是这样的



状态栏也会跟着一起滑动

## 2.其他的解决办法

## (2.1)第一种思路

自定义一个状态栏，不能添加“ android:fitsSystemWindows="true"这个属性，不然无法填充到状态栏，如下

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

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="@color/colorAccent"

android:orientation="vertical">

<View

android:id="@+id/status\_bar"

android:layout\_width="match\_parent"

android:layout\_height="20dp"/>

<FrameLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"/>\

</LinearLayout>

在到代码中判断

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

View statusBar = findViewById(R.id.status\_bar);

setContentView(R.layout.activity\_test);

//判断SDK版本是否大于等于19，大于就让他显示，小于就要隐藏，不然低版本会多出来一个

if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.KITKAT) {

setTranslucentStatus(true);

statusBar.setVisibility(View.VISIBLE);

//还有设置View的高度，因为每个型号的手机状态栏高度都不相同

}else{

statusBar.setVisibility(View.GONE);

}

}

@TargetApi(19)

private void setTranslucentStatus(boolean on) {

Window win = getWindow();

WindowManager.LayoutParams winParams = win.getAttributes();

final int bits = WindowManager.LayoutParams.FLAG\_TRANSLUCENT\_STATUS;

if (on) {

winParams.flags |= bits;

} else {

winParams.flags &= ~bits;

}

win.setAttributes(winParams);

}~~~

其实，这样已经解决了我上面提出的两个问题

2.2第二种实现方案

第二种方案是为了解决第一种方案中遇到的奇葩问题，设置了透明属性的界面（聊天及底下评论的框框）不能被系统输入法顶上去，之前写过一篇[Android 聊天界面背景图片被输入法“顶上去”问题解析](http://www.jianshu.com/p/c632fa3e44a1)，现在遇到的就是无论如何聊天的输入框都不能被系统输入法顶上去（就是打字看不到输入框），经过一番测试，发现竟然和“ android:fitsSystemWindows="true" 这个属性有关，加上去输入框就没问题，但自定义的状态栏不能被填充到真正的状态栏位置

陷入了两难的境地，加还是不加都有问题，而且都特别明显，说了半天，来看看第二种方案。

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout  
xmlns:android="[http://schemas.android.com/apk/res/android](https://link.jianshu.com?t=http:/schemas.android.com/apk/res/android)"  
xmlns:tools="[http://schemas.android.com/tools](https://link.jianshu.com?t=http:/schemas.android.com/tools)"  
android:layout\_width="match\_parent"  
android:layout\_height="match\_parent"  
android:background="@android:color/holo\_green\_light"  
android:fitsSystemWindows="true"  
tools:context="com.saidtx.myapplication.TestActivity">

<ScrollView

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_above="@+id/edit"

android:background="@android:color/white">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content">

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="center"

android:padding="20dp"

android:text="@string/previews"/>

</LinearLayout>

</ScrollView>

<LinearLayout

android:id="@+id/edit"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_alignParentBottom="true"

android:background="@android:color/white">

<EditText

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"/>

</LinearLayout>

</RelativeLayout>

关键在于下面两个属性，还有需要在其他子布局添加背景，不然就跟随了最外层的背景，代码部分还是采用网上通用方案，只是不需要自定义的状态栏了，也不需要计算状态栏的高度

android:fitsSystemWindows="true"  
android:background="@android:color/holo\_green\_light"

源码下载：[StatusBarDemo](https://github.com/dongjunkun/StatusBarDemo)

2.3 软键盘在全屏下的布局计算问题

原因是因为设置状态栏透明后没有设置android:fitsSystemWindows="true"这个属性。我的解决办法是使用[AndroidBug5497Workaround](https://link.jianshu.com?t=http:/blog.csdn.net/a2758963/article/details/25280507?utm_source=tuicool&utm_medium=referral" \t "_blank) 类动态计算布局的高度。这个方法原始代码在华为等手机上存在高度计算不准确的兼容性问题。我在源码的基础上做了一些修改，目前在华为，小米，三星等手机上测试均正常，修改后的代码如下。

public class AndroidBug5497Workaround {

public static void assistActivity(Activity activity) {

new AndroidBug5497Workaround(activity);

}

private View mChildOfContent;

private int usableHeightPrevious;

private FrameLayout.LayoutParams frameLayoutParams;

private int contentHeight;

private boolean isfirst = true;

private Activity activity;

private int statusBarHeight;

private AndroidBug5497Workaround(Activity activity) {

//获取状态栏的高度

int resourceId = activity.getResources().getIdentifier("status\_bar\_height", "dimen", "android");

statusBarHeight = activity.getResources().getDimensionPixelSize(resourceId);

this.activity = activity;

FrameLayout content = (FrameLayout)activity.findViewById(android.R.id.content);

mChildOfContent = content.getChildAt(0);

//界面出现变动都会调用这个监听事件

mChildOfContent.getViewTreeObserver().addOnGlobalLayoutListener(new ViewTreeObserver.OnGlobalLayoutListener() {

public void onGlobalLayout() {

if (isfirst) {

contentHeight = mChildOfContent.getHeight();//兼容华为等机型

isfirst = false;

}

possiblyResizeChildOfContent();

}

});

frameLayoutParams = (FrameLayout.LayoutParams)

mChildOfContent.getLayoutParams();

}

//重新调整跟布局的高度

private void possiblyResizeChildOfContent() {

int usableHeightNow = computeUsableHeight();

//当前可见高度和上一次可见高度不一致 布局变动

if (usableHeightNow != usableHeightPrevious) {

//int usableHeightSansKeyboard2 = mChildOfContent.getHeight();//兼容华为等机型

int usableHeightSansKeyboard = mChildOfContent.getRootView().getHeight();

int heightDifference = usableHeightSansKeyboard - usableHeightNow;

if (heightDifference > (usableHeightSansKeyboard / 4)) {

// keyboard probably just became visible

if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.KITKAT){

//frameLayoutParams.height = usableHeightSansKeyboard - heightDifference;

frameLayoutParams.height = usableHeightSansKeyboard - heightDifference + statusBarHeight;

} else {

frameLayoutParams.height = usableHeightSansKeyboard -heightDifference;

}

} else {

frameLayoutParams.height = contentHeight;

}

mChildOfContent.requestLayout();

usableHeightPrevious = usableHeightNow;

}

}

/\*\* \* 计算mChildOfContent可见高度 \*\* @return \*/

private int computeUsableHeight() {

Rect r = new Rect();

mChildOfContent.getWindowVisibleDisplayFrame(r);

return (r.bottom - r.top);

}

}

注意事项：  
使用这个方法有个限制是布局内得包含有大小可调的布局如ListView,ScrollView等，并且最大高度(使用权重weight)至少得大于软键盘和输入框的高度之和。一般的聊天界面本身就是个ListView，所有不用担心这个问题。

使用方法

在你的Activity的oncreate()方法里调用AndroidBug5497Workaround.assistActivity(this);即可。注意：在setContentView(R.layout.xxx)之后调用。

关于AndroidBug5497Workaround更详细的介绍可以看这里[android 软键盘在全屏下的布局计算问题](https://link.jianshu.com?t=http:/blog.csdn.net/a2758963/article/details/25280507?utm_source=tuicool&utm_medium=referral)  
链接：<https://www.jianshu.com/p/a95a1b84da11>