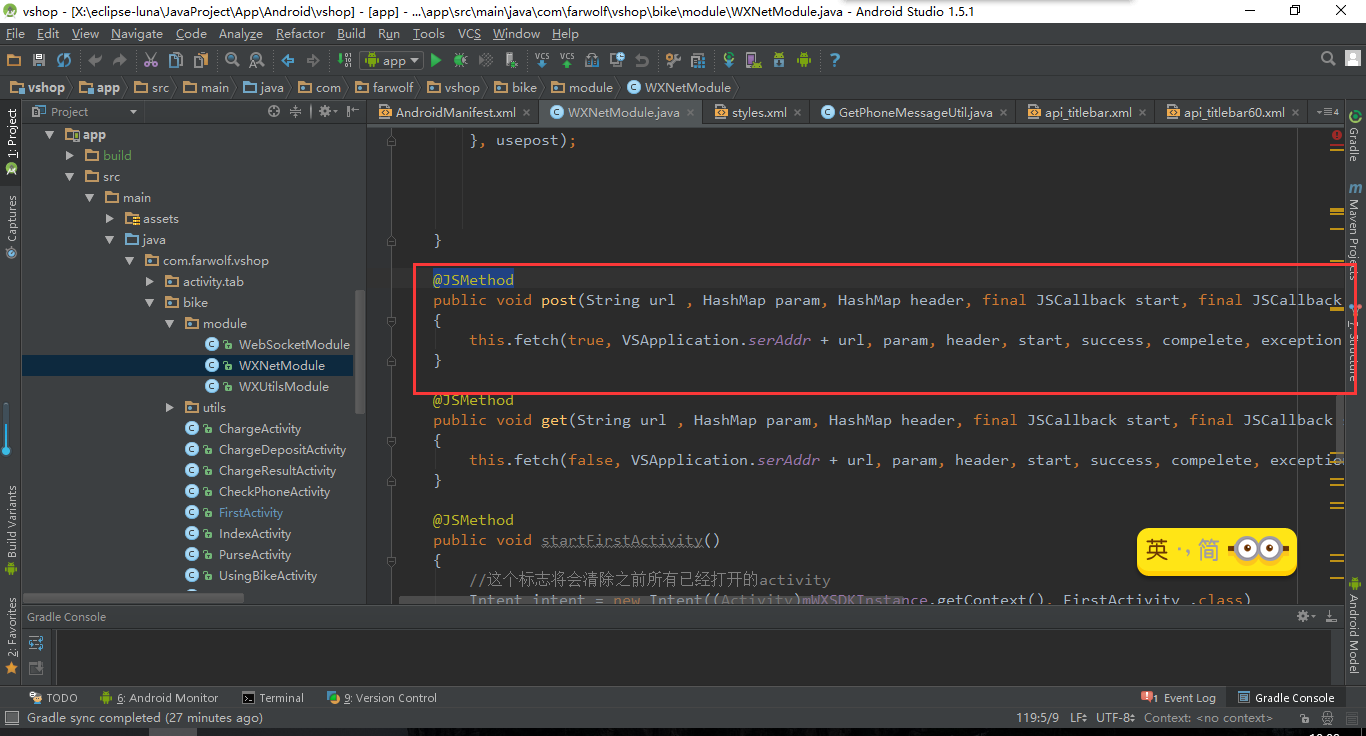
# farwolf开发文档

## farwolf自定义模块的使用

以WXNetModule为例

WXNetModule继承WXModule类，方法上有注解@JSMethod的就可以在页面上调用

在android里



在页面上调用



## activity加载js的方法

继承weexActivity

@EActivity  
public class ChargeActivity extends WeexActivity {  
  
 @Override  
 public void init() {  
 this.title.setTitle("充 值");  
 this.title.setLeftImage(R.drawable.*fanhui*);  
 this.title.setBack();  
  
 render("charge.js", false);  
 }  
}

## android调用自定义模块

new WXStorageModule().setItem("IMEI", IMEI, null);

new WXStorageModule().getItem("IMEI", new JSCallback() {

@Override

public void invoke(Object data) {

Map map = (HashMap) data;

IMEI = map.get("data") + "";

Log.i("jinming", "storage: IMEI" + map.get("data") + "");

}

@Override

public void invokeAndKeepAlive(Object data) {

}

});

## vue加载本地图片

导入LoadLocalImageUtil类

在ImageAdapter类的setImage方法中添加

if (url.contains("/assets/")) {

Log.i("com.jinming", "------------url:" + url);

LoadLocalImageUtil.getInstance().dispalyFromAssets(url.replace("/assets/",""), view);

return;

}

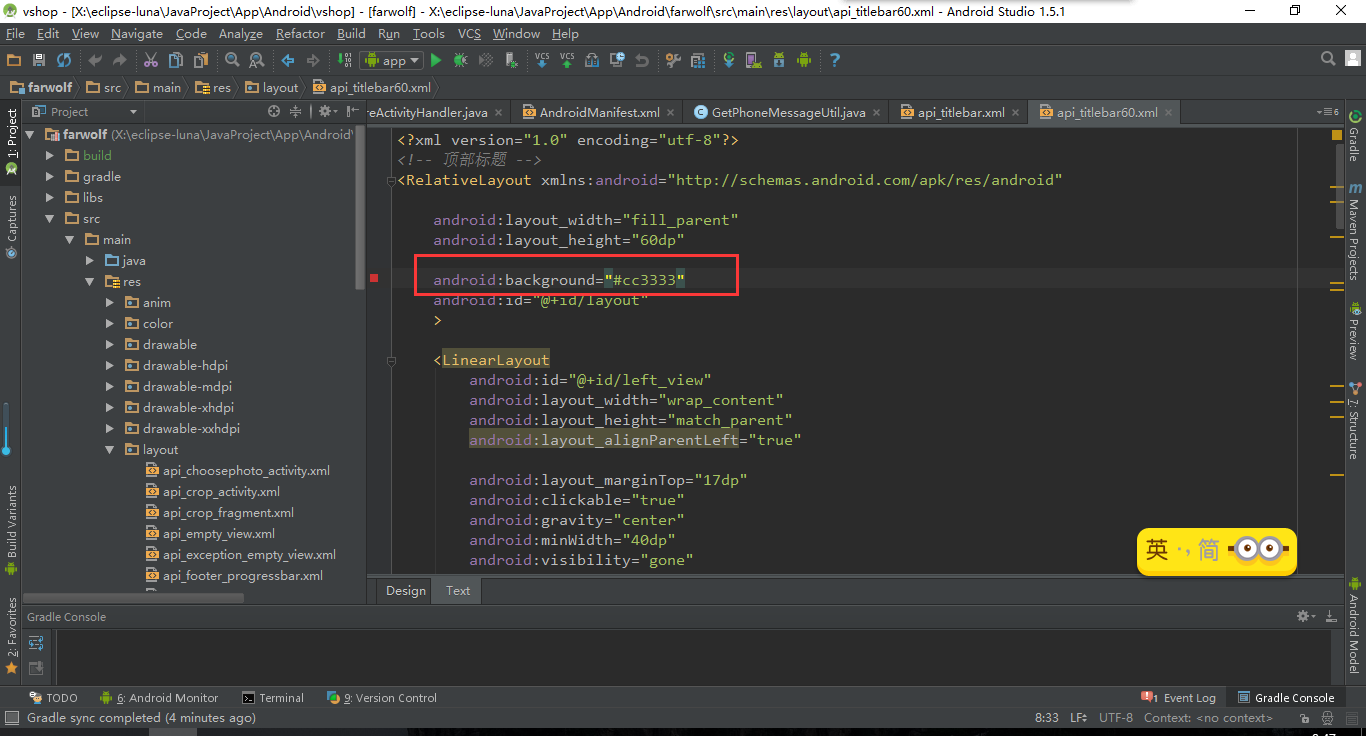
在vsapplication类中添加

WXSDKEngine.*initialize*(this,  
 new InitConfig.Builder()  
 .setImgAdapter(new ImageAdapter())  
 .build()  
);

## 标题栏的使用

### 如何修改标题栏颜色

在farwolf\src\main\res\layout\api\_titlebar60.xml和api\_titlebar.xml路径下修改android:background="#cc3333"



### 如何修改标题栏左右的点击事件

在FirstActivity的init方法里设置this.title.setLeftClick和

this.title.setRightClick

### 如何在vue文件里设置标题栏

created:function(){

var globalEvent = weex.requireModule('globalEvent') ;

var self = this;

globalEvent.addEventListener("onPageInit", function (e) {

const nav = weex.requireModule('navbar');

nav.setTitle('充 值');

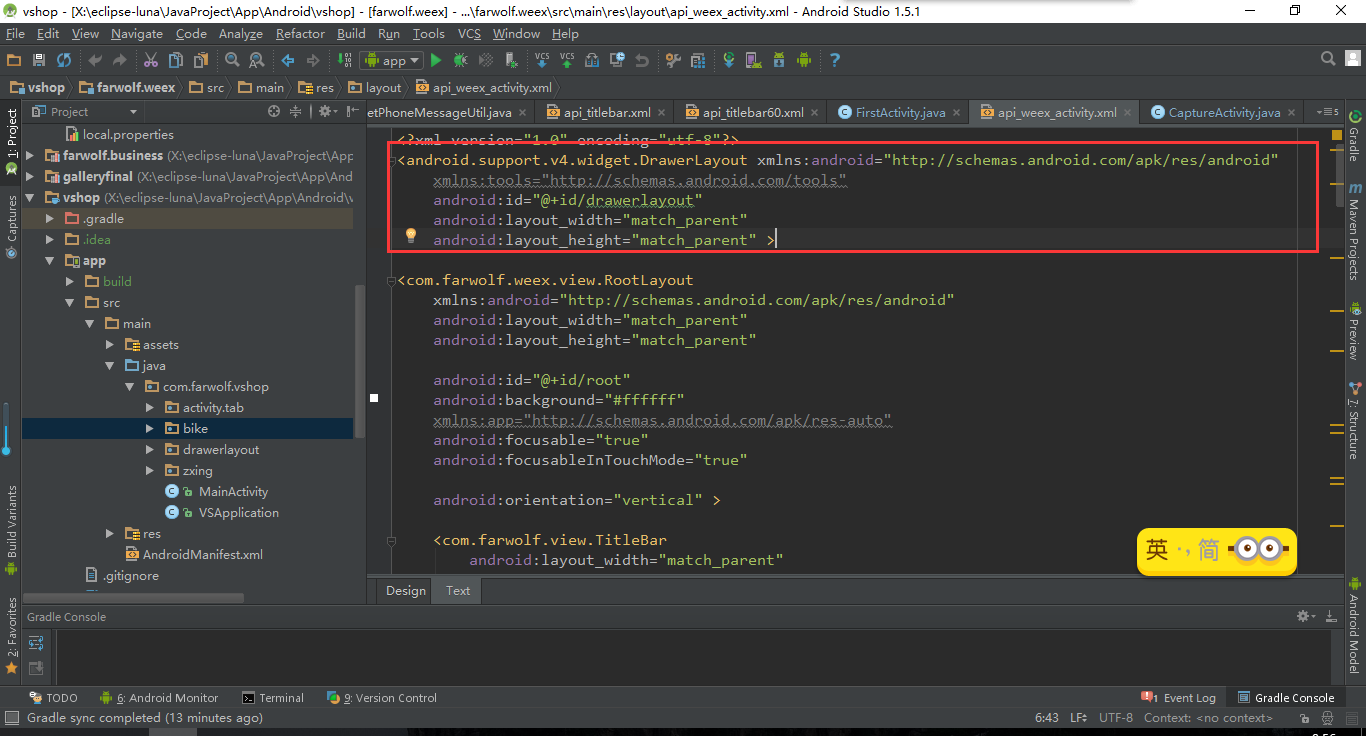
nav.setBack(true);

});

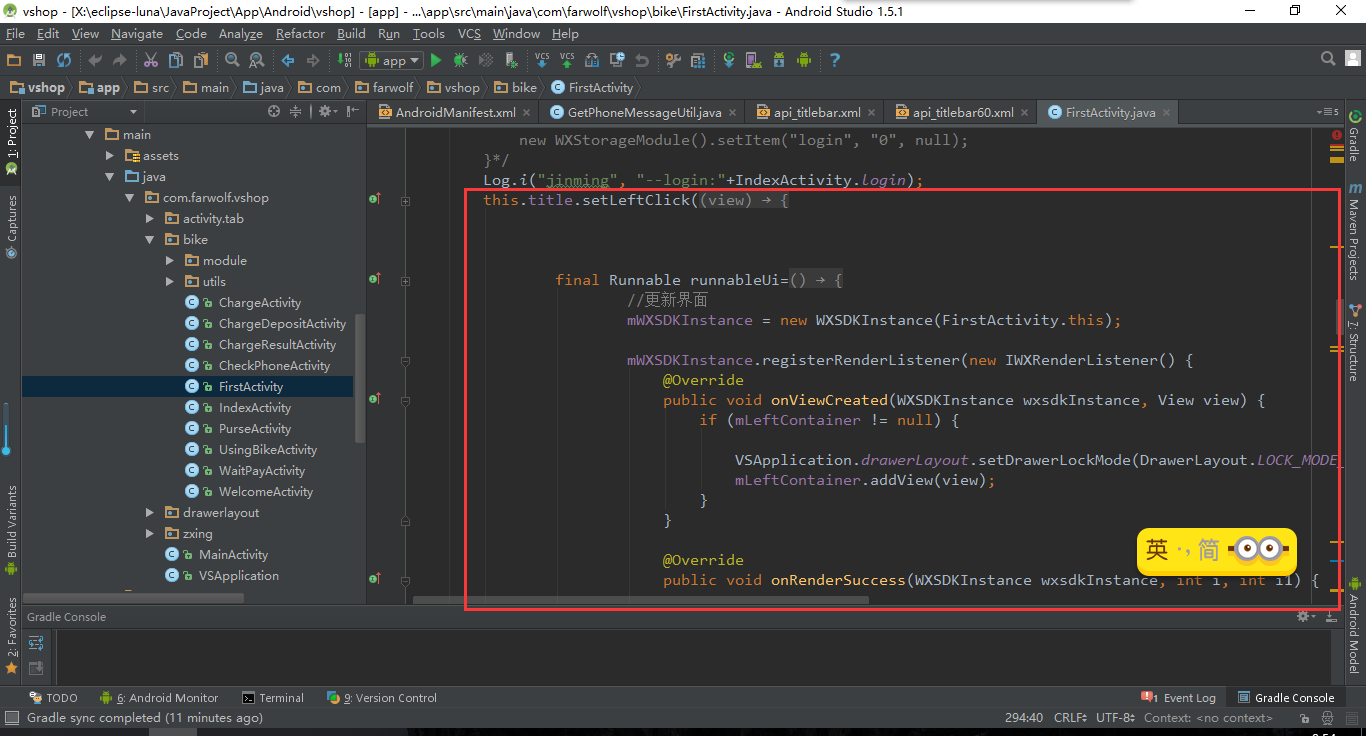
}

### 如何添加侧滑菜单栏

在farwolf.weex\src\main\res\layout\api\_weex\_activity.xml中添加drawerlayout

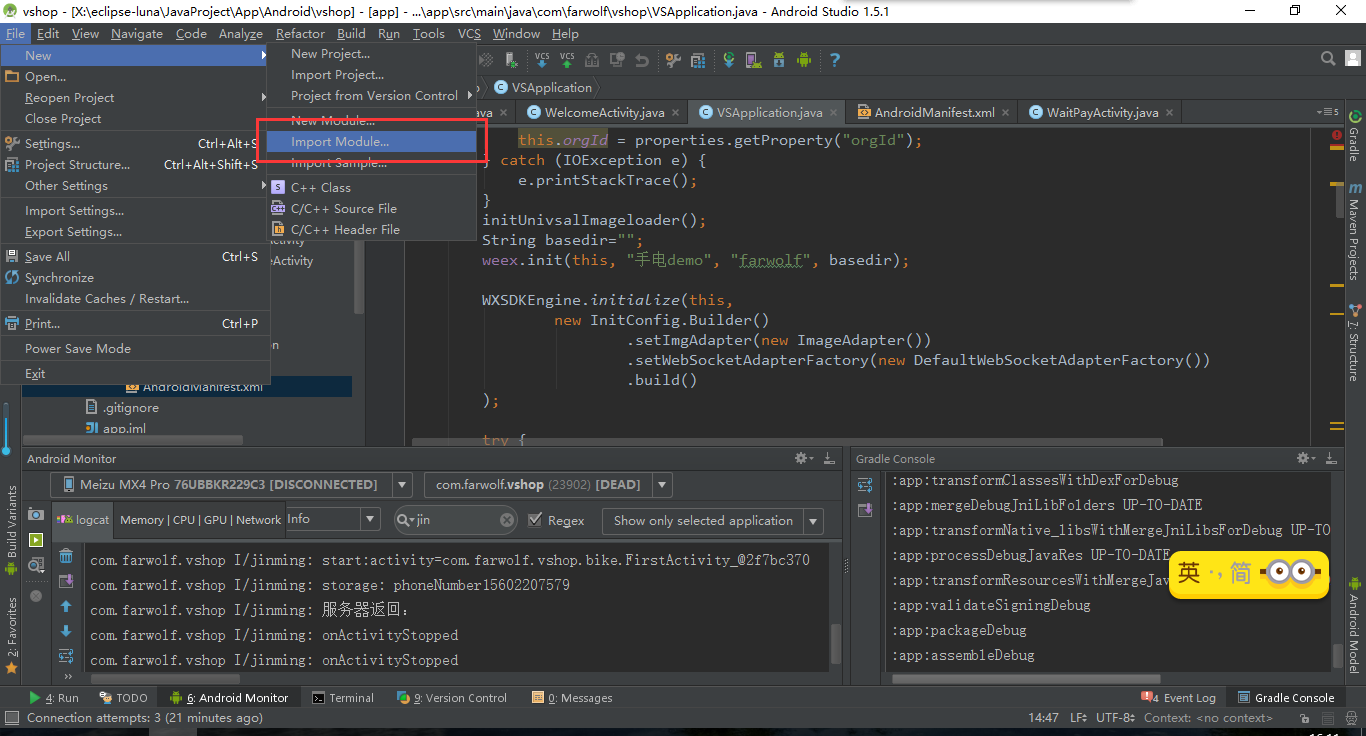


在FirstActivity的init方法中添加

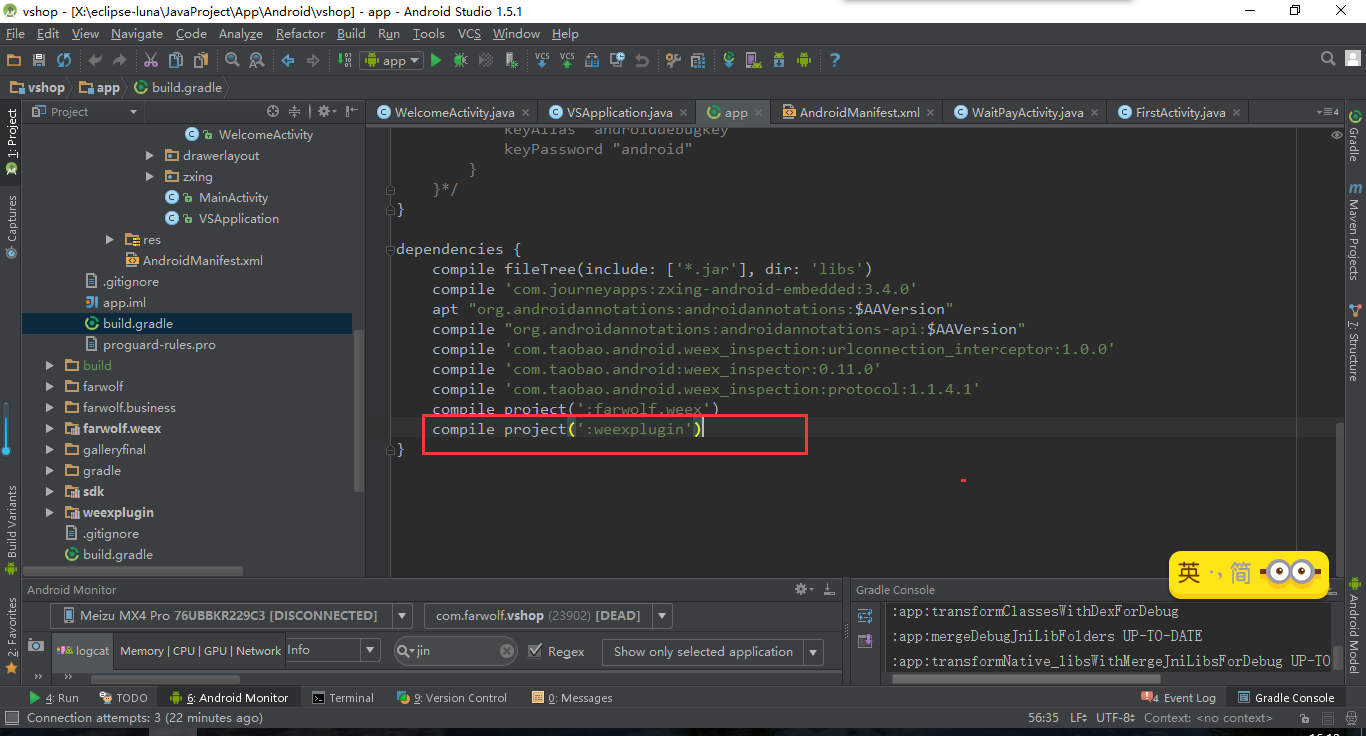


## 高德地图插件整合

导入weexplugin模块

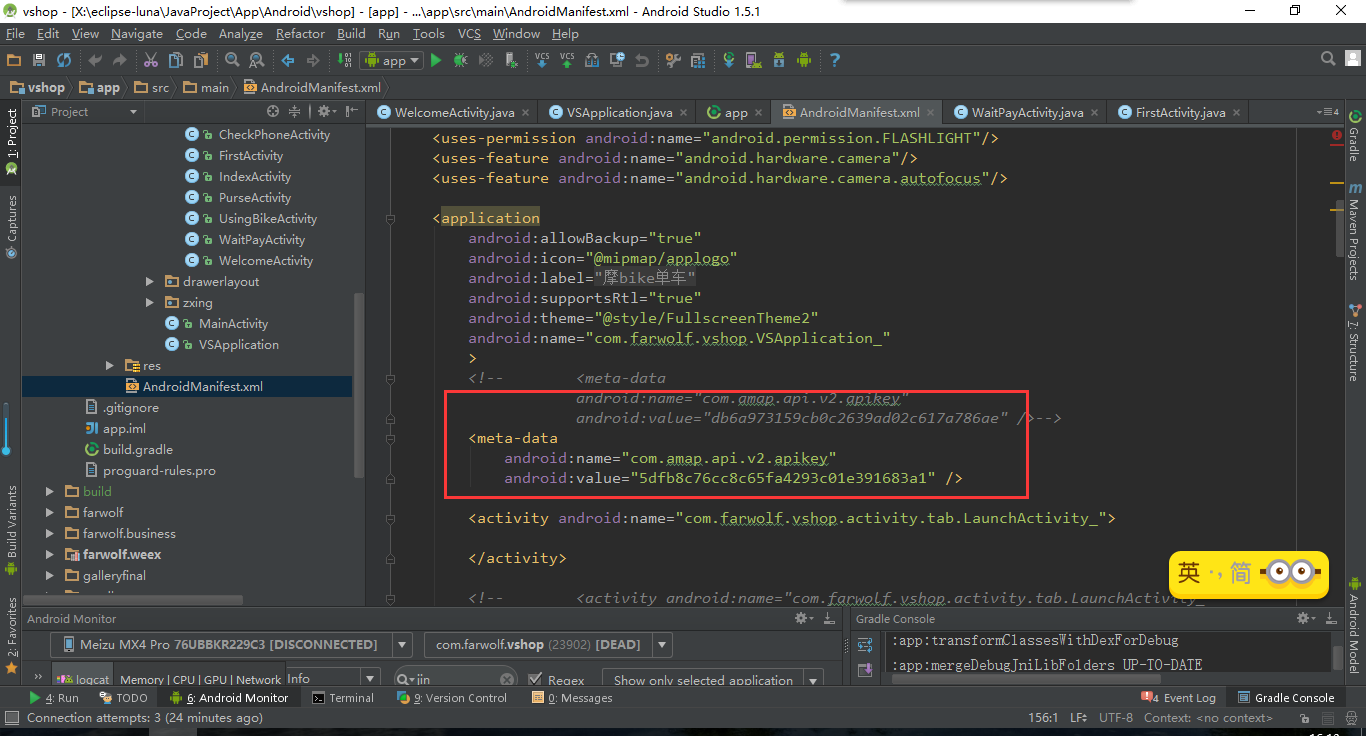


在build.gradle文件中加入compile project(':weexplugin')



在AndroidManifest.xml添加高德地图的key，需要在官网里申请一个key

<meta-data  
 android:name="com.amap.api.v2.apikey"  
 android:value="5dfb8c76cc8c65fa4293c01e391683a1" />



在AndroidManifest.xml添加地图所需的权限

//地图包、搜索包需要的基础权限

<!--允许程序打开网络套接字-->

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

<!--允许程序设置内置sd卡的写权限-->

<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />

<!--允许程序获取网络状态-->

<uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" />

<!--允许程序访问WiFi网络信息-->

<uses-permission android:name="android.permission.ACCESS\_WIFI\_STATE" />

<!--允许程序读写手机状态和身份-->

<uses-permission android:name="android.permission.READ\_PHONE\_STATE" />

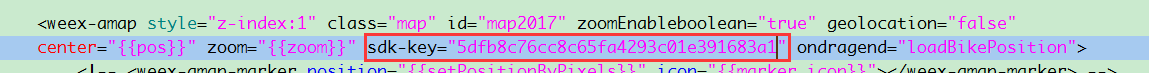
<!--允许程序访问CellID或WiFi热点来获取粗略的位置-->

<uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION" />

## 高德地图定位key的设置

高德地图申请key的方法：<http://lbs.amap.com/api/android-sdk/guide/create-project/get-key>

申请后在weex-amap组件添加



在AndroidManifest.xml添加

<meta-data  
 android:name="com.amap.api.v2.apikey"  
 android:value="5dfb8c76cc8c65fa4293c01e391683a1" />

## 权限的添加

android6.0以上要自己在activity中手动添加权限

if (ActivityCompat.*checkSelfPermission*(FirstActivity.this, Manifest.permission.*READ\_PHONE\_STATE*) != PackageManager.*PERMISSION\_GRANTED*) {  
 // toast("需要动态获取权限");  
 ActivityCompat.*requestPermissions*(FirstActivity.this, new String[]{Manifest.permission.*READ\_PHONE\_STATE*}, REQUEST\_PHONE\_STATE);  
}

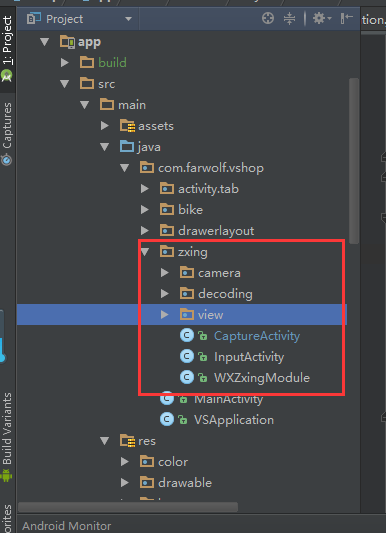
*/\*\*  
 \*加个获取权限的监听  
 \*/*@Override  
public void onRequestPermissionsResult(int requestCode, String[] permissions,int[] grantResults) {  
 if (requestCode == REQUEST\_PHONE\_STATE && grantResults.length == 1 && grantResults[0] == PackageManager.*PERMISSION\_GRANTED*) {  
  
 }  
}

## 扫码插件整合

在build.gradle中添加

compile 'com.journeyapps:zxing-android-embedded:3.4.0'

在src/java下添加



### 如何添加闪光灯

在androidmanifest.xml中添加权限

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

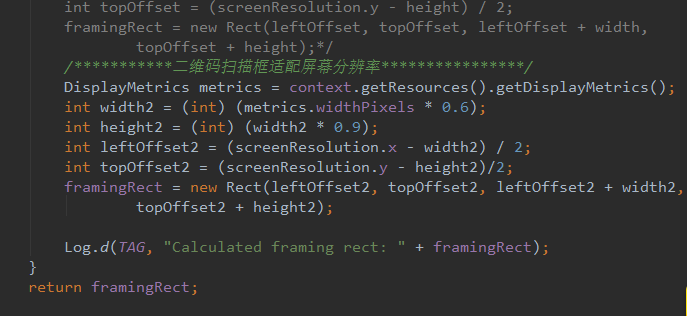
<uses-permission android:name="android.permission.CAMERA" />

在captureActivity中添加关键代码

*/\*\*  
 \* 开灯  
 \*/*private OnClickListener openListener = new OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 camera = CameraManager.*getCamera*();  
 parameter = camera.getParameters();  
 // *TODO 开灯* if (isOpen) {  
 //btnOpen.setText("关灯");  
 parameter.setFlashMode(Parameters.*FLASH\_MODE\_TORCH*);  
 camera.setParameters(parameter);  
 isOpen = false;  
 } else { // 关灯  
 //btnOpen.setText("开灯");  
 parameter.setFlashMode(Parameters.*FLASH\_MODE\_OFF*);  
 camera.setParameters(parameter);  
 isOpen = true;  
 }  
 }  
};

### 扫码框如何自适应屏幕

在cameraManager类的getFramingRect中添加



### 扫码如何将竖屏改为横屏、不拉伸

<http://www.cnblogs.com/moka/archive/2013/05/24/3096937.html>

只记得关键代码

在CameraConfigurationManager类的initFromCameraParameters方法中添加

Camera.Parameters parameters = camera.getParameters();

previewFormat = parameters.getPreviewFormat();

previewFormatString = parameters.get("preview-format");

WindowManager manager = (WindowManager)context.getSystemService(Context.WINDOW\_SERVICE);

Display display = manager.getDefaultDisplay();

int width = display.getWidth();

int height = display.getHeight();

screenResolution = new Point(width,height);

if(width < height)

{

cameraResolution = getCameraResolution(parameters, new Point(height,width));

}else{

cameraResolution = getCameraResolution(parameters, screenResolution);

}

### 扫码界面如何自定义标题栏

相关布局文件：zxing.xml和pub\_titlebar.xml

在zxing.xml添加

<com.farwolf.vshop.zxing.view.CustomTitleBar  
 android:id="@+id/ct"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 app:title\_text="扫描开锁"  
 app:title\_textColor="#ffffff"  
 app:title\_background="#cc3333"  
 app:right\_button\_text="扫码帮助"  
 app:right\_button\_textColor="#ffffff"  
 >  
  
</com.farwolf.vshop.zxing.view.CustomTitleBar>

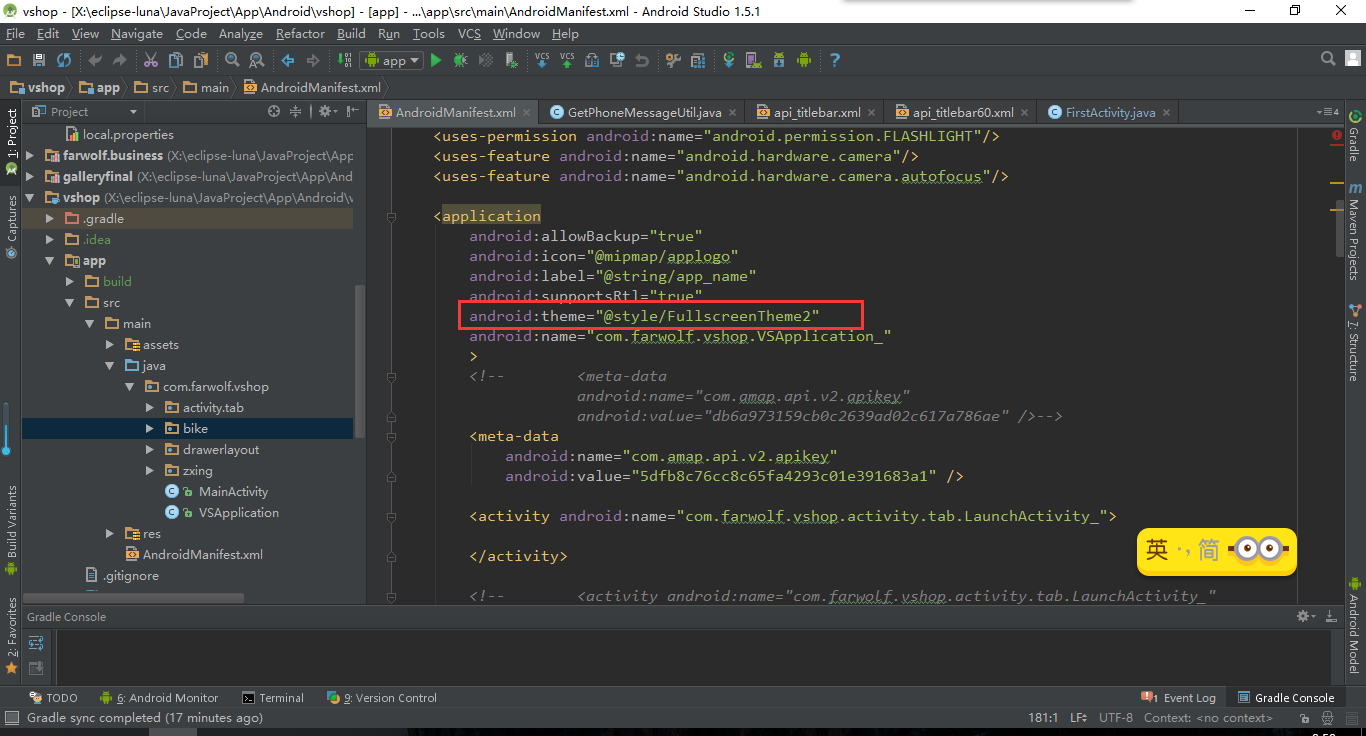
在pub\_titlebar.xml中可修改标题栏的字体、图片等大小

## 欢迎界面的修改

在styles.xml文件中添加

<style name="FullscreenTheme2" parent="AppTheme">  
 <item name="windowActionBar">false</item>  
 <item name="windowNoTitle">true</item>  
 <item name="android:windowActionBarOverlay">true</item>  
 <item name="android:windowBackground">@drawable/bg1</item>  
 <item name="android:windowFullscreen">true</item>  
</style>

在AndroidManifest文件中的application添加android:theme="@style/FullscreenTheme2"



## app名称和图标的修改

在AndroidManifest文件中的application添加

android:icon="@mipmap/applogo"  
android:label="@string/app\_name"

