

# YiCameraSDK 文档说明-Android

## 1. 简介

该SDK主要包括以下功能:

1. 绑定设备
2. 设备列表
3. 设备报警列表
4. 视频直播
5. 设备设置
6. 设备云存

## 2. 集成依赖库

复制yicamerasdk.aar到您项目工程中libs目录下, 在build.gradle中添加以下依赖库:

```
implementation fileTree (include: ['* .jar', '* .aar'], dir: 'libs')
implementation "androidx.cardview:cardview:1.0.0"
implementation 'com.google.android.material:material:1.0.0'
implementation 'androidx.multidex:multidex:2.0.1'
implementation 'com.squareup.retrofit2:retrofit:2.5.0'
implementation 'com.squareup.retrofit2:adapter-rxjava2:2.5.0'
implementation 'com.squareup.retrofit2:converter-gson:2.4.0'
implementation 'com.squareup.okhttp3:logging-interceptor:3.12.0'
implementation 'com.google.code.gson:gson:2.8.5'
implementation 'com.google.zxing:core:3.3.0'
implementation "androidx.core:core-ktx:+"
implementation "org.jetbrains.kotlin:kotlin-stdlib-jdk7:$kotlin_version"

implementation 'androidx.multidex:multidex:2.0.1'
api 'com.squareup.retrofit2:retrofit:2.5.0'
api 'com.squareup.retrofit2:adapter-rxjava2:2.5.0'
api 'com.squareup.retrofit2:converter-gson:2.4.0'
implementation 'io.reactivex.rxjava2:rxjava:2.2.8'
implementation 'io.reactivex.rxjava2:rxandroid:2.1.1'
implementation 'com.squareup.okhttp3:okhttp:3.12.0'
implementation 'com.iheartradio.m3u8:open-m3u8:0.2.4'
implementation 'androidx.appcompat:appcompat:1.1.0'
implementation 'androidx.recyclerview:recyclerview:1.1.0'
implementation "androidx.cardview:cardview:1.0.0"

implementation 'com.google.dagger:dagger:2.24'

implementation 'androidx.paging:paging-rxjava2-ktx:2.1.0'
implementation 'androidx.paging:paging-common-ktx:2.1.0'
implementation 'androidx.paging:paging-runtime-ktx:2.1.0'
implementation 'com.squareup.okhttp3:logging-interceptor:3.12.0'

implementation 'com.uber.autodispose:autodispose-android:1.1.0'
implementation 'com.uber.autodispose:autodispose-ktx:1.1.0'
implementation 'com.uber.autodispose:autodispose-android-ktx:1.1.0'
implementation 'com.uber.autodispose:autodispose-android-archcomponents-ktx:1.1.0'

implementation 'androidx.room:room-runtime:2.1.0-alpha06'
implementation 'androidx.room:room-ktx:2.1.0-alpha06'
annotationProcessor 'androidx.room:room-compiler:2.1.0-alpha06'
kapt 'androidx.room:room-compiler:2.1.0-alpha06'
implementation 'androidx.room:room-rxjava2:2.1.0-alpha06'
implementation 'androidx.room:room-testing:2.1.0-alpha06'
implementation 'androidx.lifecycle:lifecycle-viewmodel:2.0.0'
implementation 'androidx.lifecycle:lifecycle-extensions:2.0.0'

implementation 'com.github.bumptech.glide:glide:4.9.0'
kapt 'com.github.bumptech.glide:compiler:4.9.0'
implementation 'com.alibaba:arouter-api:1.5.0'
kapt 'com.alibaba:arouter-compiler:1.2.2'
```

## 3. 集成YiCameraSdk步骤简介

首先, 必须先设置appid和用户信息, 用户信息包括openid, usertoken, 用户信息可从小蚁获取。 然后, 可以按集成Activity或Fragment方式集成相关界面。

## 4. 初始化YiCameraSdk

```
// YiCameraSDK
YiCameraSdk.init(getApplication(),appid);
//
YiCameraSdk.setUserInfo(openid,token);

setUserInfo:
1.YiCameraSdk.setOnUserVerifyListener(this);
2.Make your activity implements YiCameraSdk.OnUserVerifyListener
3.Add the following method:
@Override
public void OnUserVerifyResult(boolean result,int code) { //result: true , false
    //
}
```

## 5. 以Activity方式集成

### 打开设备绑定界面

```
YiCameraSdk.openBindChoiceModeActivity(Context context);

Activity:
1.YiCameraSdk.setOnBindDeviceListener(this);
2.Make your activity implements YiCameraSdk.OnBindDeviceListener
3.Add the following method:
@Override
public void OnBindDeviceResult(boolean result, String uid, String name) { //result: true , false
    //
}
```

### 打开设备报警列表界面

```
YiCameraSdk.openNewAlertActivity(Context context);
```

### 打开设备列表界面

```
YiCameraSdk.openDeviceListActivity(Context context);
```

### 打开直播视频界面

```
YiCameraSdk.openCameraPlayerActivity(uid, new YiCameraSdk.YICallback() { //uidid
    @Override
    public void onResult(int code, String result) {

    }
});
```

### 打开设备设置界面

```
YiCameraSdk.openCameraSettingActivity(uid, new YiCameraSdk.YICallback() { //uidid
    @Override
    public void onResult(int code, String result) {

    }
});
```

### 打开云存播放界面

```
YiCameraSdk.openCloudVideoActivity(Context context);

upgrade:
1.YiCameraSdk.setOnUpgradeClickedListener(this);
2.Make your activity implements YiCameraSdk.OnUpgradeClickedListener
3.Add the following method:
@Override
public void onUpgradeClicked(String deviceId, String orderCode, int currentStatus) {
    //
}
```

## 6. 以Fragment方式集成

### 打开设备绑定界面

```

OpenChoiceModeFragment fragment = new OpenChoiceModeFragment();
FragmentManager ft = getSupportFragmentManager().beginTransaction();
ft.add(containerId, fragment);
ft.commitAllowingStateLoss();

Activity:
1.YiCameraSdk.setOnBindDeviceListener(this);
2.Make your activity implements YiCameraSdk.OnBindDeviceListener
3.Add the following method:
@Override
public void onBindDeviceResult(boolean result, String uid, String name) { //result: true , false
    //
}

```

### 打开设备报警列表界面

```

NewAlertFragment fragment = new NewAlertFragment();
FragmentManager ft = getSupportFragmentManager().beginTransaction();
ft.add(containerId, fragment);
ft.commitAllowingStateLoss();

```

### 打开设备列表界面

```

DeviceListFragment deviceListFragment = new DeviceListFragment();
FragmentManager ft = getSupportFragmentManager().beginTransaction();
ft.add(containerId, deviceListFragment);
ft.commitAllowingStateLoss();

```

### 打开直播视频界面

```

StartCameraPlayerFragment fragment = StartCameraPlayerFragment.newInstance(uid, new YiCameraSdk.
YICallback() { //uid is the unique id of device
    @Override
    public void onResult(int code, String result) {

    }
});
FragmentManager ft = getSupportFragmentManager().beginTransaction();
ft.add(containerId, fragment);
ft.commitAllowingStateLoss();

```

### 打开设备设置界面

```

CameraSettingFragment fragment = CameraSettingFragment.newInstance(uid);
//uid is the unique id of device
FragmentManager ft = getSupportFragmentManager().beginTransaction();
ft.add(containerId, fragment);
ft.commitAllowingStateLoss();

```

### 打开云存播放界面

```

CloudVideoFragment cloudVideoFragment = new CloudVideoFragment();
FragmentManager ft = getSupportFragmentManager().beginTransaction();
ft.add(containerId, cloudVideoFragment);
ft.commitAllowingStateLoss();

upgrade:
1.YiCameraSdk.setOnUpgradeClickedListener(this);
2.Make your activity implements YiCameraSdk.OnUpgradeClickedListener
3.Add the following method:
@Override
public void onUpgradeClicked(String deviceId, String orderCode, int currentStatus) {
    //
}

```

### 监听fragment返回按钮点击事件:

```

1.YiCameraSdk.setOnFragmentBackBtnListener(this);
2.Make your activity implements YiCameraSdk.OnFragmentBackBtnListener
3.Add the following method:
@Override
public void OnFragmentBackBtn(int type) {
    /**
     * type:
     *      YiCameraSdk.BIND_FRAGMENT:
     *      YiCameraSdk.CAMERA_PLAY_FRAGMENT:
     *      YiCameraSdk.CAMERA_SETTINGS_FRAGMENT:
     *      YiCameraSdk.CAMERA_SETTINGS_FRAGMENT_DELETE:    " "
     */

    //
}

```

## 7.API接口

```

//
getDeviceList().observeOn(AndroidSchedulers.mainThread()).subscribe(new Consumer<List<DeviceInfo>>() {
    @Override
    public void accept(List<DeviceInfo> deviceInfos) throws Exception {
        //Return device list data
    }, new Consumer<Throwable>() {
    @Override
    public void accept(Throwable throwable) throws Exception {
        //Return Error
    }
});

// 4Giccid
YiCameraSdk.get4gInfoWithUid(String uid, new YiCameraSdk.Callback() {
    @Override
    public void onResult(boolean result, String iccid) {

    }
});

```

## 8.版本记录

更新记录	平台	版本	发布日期
添加绑定结果回调	Android SDK	V1.2.1_20201214	2020/12/14
初始发布版本	Android SDK	V1.2.0_20201208	2020/12/08