

# 招生宝使用的 Android 依赖库

---

没有开源社区的支持,就不会有现在 Android 的繁荣,当然招生宝也离不开开源社区的共享,此文就将介绍招生宝 Android 端所使用的依赖库.

[招生宝的第三方开源协议](#)

## 测试相关

```
//test
junit:junit:4.12
org.mockito:mockito-all
org.hamcrest:hamcrest-all

// Android Testing Support Library's runner and rules
com.android.support.test:runner:0.5
androidcom.android.support.test:rules

// Dependencies for Android unit tests
androidjunit:junit:$rootProject.ext.junitVersion"
androidorg.mockito:mockito-
core:$rootProject.ext.mockitoVersion"
'com.google.dexmaker:dexmaker:1.2'
'com.google.dexmaker:dexmaker-mockito:1.2'

// Espresso UI Testing
com.android.support.test.espresso:espresso-core:2.2.2

com.android.support.test.espresso:espresso-contrib
androidcom.android.support.test.espresso:espresso-intents

// Resolve conflicts between main and test APK:
```

```
androidcom.android.support:support-annotations  
androidcom.android.support:support-v4  
androidcom.android.support:recyclerview-v7
```

这是是从 [Android Architecture Blueprints](#) 上看到而使用的, 用于 Unit 测试, UI 测试等用途

## support library

com.android.support:appcompat-v7

com.android.support:recyclerview-v7

com.android.support:design

com.android.support:support-annotations

com.android.support:cardview-v7

com.android.support:customtabs

com.android.support:percent

Google 将 Android 的更新分为了4种,系统更新,系统应用更新,Google Play Services 更新和 support library 更新. 系统更新很慢,Android Marshmallow 已经出来了一年了,然而市占率仅有 15%.这显然达不到预期. 剩余的部分更新都很快速,系统应用更新和 Google Play Services 都通过 Play 进行更新,对于终端用户来说仅仅只是更新了一个应用而已. Support Library 则是由开发者来更新,使得应用能在第一时间使用部分 Android 的新特性. 同时也推动着开发者去适配新的系统平台,因为只有适配了最新的系统,才能使用最新的 Support Library.

appcompat 是一个兼容库,提供了 actionbar,appcompat Activity 和部分 MD 用户界面.

design,cardview-v7 提供了另一些 MD 的用户界面.

recyclerview 替代了 listview,成为了显示列表的首选.

support-annotations 提供了 annotation 例如 intdef,nonnull,nullable等等用于提

升代码质量的声明.

customtabs 让应用能够使用 chrome 内核的特性,不用在忍受旧系统的辣鸡 webview 了.

percent 是百分比视图库, 百分比视图在 iOS 设备上还是用的比较多的,如果使用 iOS 转制的设计图,那么这个库我想是避不开了.

## UI 库

com.leavjenn.smoothdaterangepicker

[SmoothDateRangePicker](#) 这是一个 MD 的时间范围选择器,不过不支持 v4 support fragment ,所以需要下载原始代码进行修改(也就是把 fragment 换到 support v4 下就好了).

com.roughike:bottom-bar

MD 在 6 月份引入了 iOS 平台和国内安卓大量使用的底部导航, [这个库](#)即提供了较为标准的 MD 底部导航 UI .

com.github.PhilJay:MPAndroidChart

[MPAndroidChart](#) 是 Android 平台著名的图表库, 提供了大量图表的解决方案.

com.makeramen:roundedimageview

圆形头像是目前很常见的头像展示方式, [roundedimageview](#) 就提供了一个包括圆角矩形,椭圆的圆形的 imageview.

com.soundcloud.android:android-crop

图片裁剪在国产设备上到处都是坑,[android-crop](#) 就提供了一个图片裁剪的 UI,使得不同设备都能够进行图片裁剪,虽然这会无法使用系统的裁剪视图.

## 网络库

com.github.bumptech.glide

[glide](#) 是一个图片加载库,支持 download only 获取缓存文件.

com.squareup.retrofit2

com.squareup.okhttp3

[retrofit](#) 是一个 restFul 客户端. retrofit 底层使用的是 [okhttp](#) ,支持 HTTP/2.

## 代码质量

com.squareup.leakcanary

[leakcanary](#) 这是一个检查内存泄漏的库,能够非常方便地追踪应用内部存在内存泄漏的部分.

io.reactivex:rxandroid

io.reactivex:rxjava

com.tbruyelle.rxpermissions:rxpermissions

com.jakewharton.rxbinding

rxjava 为 Android 开发者带来了简洁的异步库和事件流库,再也不会被 AsyncTask,Thread所困扰了w

com.facebook.stetho

[stetho](#) 是一个调试桥,可以在 chrome 终端调试应用,获取应用的网络请求等等.

me.yokeyword:fragmentation

Fragment 的管理一直是 Android 的一大难点,特别是因为内存低被系统杀死后的恢复,见过多个应用都掉进了这个大坑. [fragmentation](#) 就提供了一个包装,让开发者不用再纠结与 fragment 的恢复.

com.rockerhieu:rv-adapter

recyclerview 已经替换了 listview ,然而列表的空状态,错误状态,加载状态什么额度要怎么实现呢? 为每一个 adapter 添加状态的 view? [rv-adapter](#) 提供了对 adapter 的一个包装,你可以在任何一个 recyclerview 的 adapter 外添加包装,以支持各种状态或上划刷新等等实现.

me.xuender:unidecode

应用使用了 umeng 的统计,然而 umeng 统计不支持中文事件,还必须要求上传事件 ID 与名称对应表. 哪有时间弄这种玩意,于是直接使用 [unidecode](#) 将中文转成拼音作为事件 ID,这个库非常小巧,只有26个方法,dex 只有 3 KiB.