Android是一种基于Linux的自由及开放源代码的操作系统，主要使用于移动设备，如智能手机和平板电脑，由Google公司和开放手机联盟领导及开发。

2008年9月23日 Andorid 1.0 SDK发布

2009年4月30日 Android 1.5 SDK发布，Cupcake纸杯蛋糕

2009年9月15日 Android 1.6 SDK发布，Donut甜甜圈，这时Android进入到起飞的阶段，HTC这时候牛B，如HTC Hero G3

2009年10月26日Android 2.0 SDK发布，Eclair松饼，这时的摩托罗拉因为生产了Android手机又牛了

2009年10月23日 T-Mobile G1 (Dream) 正式发售

2010年1月11日 Android 2.1 SDK发布，此时Nexus One正式发布，谷歌自己的手机品牌。

2010年5月20日 Android 2.2 SDK发布，Froyo冻酸奶

2010年12月7日 Android 2.3.x SDK 发布，Gingerbread姜饼，三星的Nexus S发布

2011年2月2日Android 3.x honeycomb蜂巢

2011年10月19日 4.0 Ice Cream Sandwich冰激凌三明治

2012年6月28日 4.1 JellyBean 果冻豆

2012年10月30日4.2 JellyBean 果冻豆

2013年下半年 4.4 KitKat 奇巧巧克力

2015年5月28日 6.0 Marshmallow 棉花糖

2016年8月22日 7.0 Nougat 牛轧糖

4.0 Ice CreamSandwich







# Android 3.0 代号Honeycomb 蜂巢

2010年10月发布

新UI： <http://www.jianliw.com/23021.html>

AdapterViewAnimator

AdapterViewAnimator（堆View），已知子类有AdapterViewFlipper, StackView，是一系列View的集合，这些View叠加一起，并且View之间可以进行切换，并且在多个View切换过程体现渐隐渐现的动画效果。通过AdapterViewAnimator.setAdapter()方法设置一组要显示的View，通过AdapterViewAnimator.showPrevious()方法移动到前一个View,通过AdapterViewAnimator.showNext()移动到下一个View

PropertyAnimation

在android 3.0之前，支持两种动画模式，Tween Animation and Frame Animation。在3.0之后引入了PropertyAnimation，为属性动画。在View Animation（Tween Animation）中，其改变的是View的绘制效果，真正的View的属性保持不变，比如无论你在对话中如何缩放Button的大小，Button的有效点击区域还是没有应用动画时的区域，其位置与大小都不变。而在Property Animation中，改变的是对象的实际属性，如Button的缩放，Button的位置与大小属性值都改变了。而且PropertyAnimation不止可以应用于View，还可以应用于任何对象。PropertyAnimation只是表示一个值在一段时间内的改变，当值改变时要做什么事情完全是你自己决定的。

使用：<http://blog.csdn.net/a15874647/article/details/7683073>

Loader

装载器从android3.0开始引进。它使得在activity或fragment中异步加载数据变得简单。装载器具有如下特性：

它们对每个Activity和Fragment都有效。

他们提供了异步加载数据的能力。

它们监视数据源的一将一动并在内容改变时传送新的结果。

当由于配置改变而被重新创建后，它们自动重连到上一个加载器的游标，所以不必重新查询数据。

学习Loader的使用：<http://blog.csdn.net/a15874647/article/details/7683330>

引入ActionBar替代传统的TitleBar，主要针对于平板开发。对于Android平板设备来说，屏幕更大使用ActionBar可以扩展更多丰富的内容，方便操控。

主要包含功能：

1）显示选项菜单

2）提供标签页的切换方式的导航功能，可以切换多个fragment

3）提供下拉的导航条目

4）提供交互式活动视图代替选项条目

5）返回主页或返回上一页的导航操作。

\*\*兼容方案：\*\*actionbarsherlock，作为library

ListPopupWindow

CalendarView

NumberPicker

PopupMenu

SearchView

Fragment，在v4包

ActionBar，在v7包

ViewPager，在v4包

# Android 4.0新增UI

GridLayout

TextureView

Switch

# Android 5.0新特性

代号Lollipop 棒棒糖

1、全新Material Design设计风格（一种基于纸页质感的设计），5000多个新增api

Android Lollipop全新的设计语言是受到了多种因素影响，是一种大胆的平面化创新。换句话说，谷歌希望能够让Material Design给用户带来纸张化的体验。新的视觉语言，在基本元素的处理上，借鉴了传统的印刷设计，字体版式、网格系统、空间、比例、配色、图像使用等这些基础的平面设计规范。

另外，Material Design还推崇实体隐喻理念，利用实体的表面与边缘的质感打造出视觉线索，让用户感受到真实。熟悉的触感让用户可以快速的理解、认知。在设计中可以灵活的运用物质，在符合物理规律的基础上，打造出不同的使用体验。最后是是有意义而且更合理的动态效果，为了吸引用户的注意力，以及维持整个系统的连续性体验。

2、支持多种设备

现在无论是智能手机、平板电脑、笔记本电脑、智能电视、汽车、智能手表甚至是各种家用电子产品，谷歌的Android系统已经可以在所有设备的屏幕上出现。而这一概念与微软不谋而合，之前微软也宣布将会把Windows 10打造成跨设备跨平台的统一系统，帮助自己走出困境。

3、全新的通知中心设计

谷歌在Android Lollipop中加入了全新风格的通知系统。改进后的通知系统会优先显示对用户来说比较重要的信息，而将不太紧急的内容隐藏起来。用户只需要向下滑动就可以查看全部的通知内容。

新的通知系统另外一个很酷的新功能是在锁屏界面也可以直接查看通知消息了。不仅如此，用户还可以直接在锁屏的情况下就行回复或进入应用。另外，如果在操作手机的过程中有电话进入，也不会进行全画面切换，而是同样以弹出通知的方式告知用户。

4、支持64位ART虚拟机

新系统不仅在视觉效果上带来了巨大的变化，Android Lollipop还在内部的性能上进行了飞跃。首先，新系统放弃了之前一直使用的Dalvik虚拟机，改用了ART模式，实现了真正的跨平台编译，在ARM、X86、MIPS等，无处不在。

ART虚拟机编译器在内存占用及应用程序加载时间上进行了大幅提升，谷歌承诺所有性能都会比原来提升一倍。另外，对64位的支持也让ART虚拟机如鱼得水，开发者可以针对像ARM Cortex-A57这样的64位架构核心开发应用程序。

Android Lollipop支持更大的寄存器，支持新的指令集，提升了内存寻址空间，未来Android智能手机将支持4GB以上的内存。

CardView

顾名思义，CardView 卡片视图，继承自framelayout，可以通过设置圆角以及阴影来展示带有像卡片一样的效果

RecyclerView的出现可以替代ListView和GridView，它标准化了ViewHolder，之前我们在写ListView的Adapter时需要自定义ViewHolder来提升ListView速度。使用RecyclerView的时候必须要设置LayoutManager，它会通知系统以什么样的布局来展示RecyclerView，目前系统提供了2种LayoutManager，LinearLayoutManager和GridLayoutManager对应着线性和格子，当然了也可以自定义LayoutManager来满足各种需求

# Android 6.0 新特性

运行时权限

去除了Apache的http Client相关方法.Google建议用httpURLConnection来替代httpclient的相关方法,如果执意继续使用httpclient 相关方法,那么你就需要自取导入Apache的jar包.在Android Studio中的Gradle里通过userlibrary'org.apache.http.legacy'去引用.

支持快速充电的切换.可以开启或关闭快速充电

# Android 7.0新特性

1.分屏多任务支持

2.画中画

3.通知栏快速回复

4.OpenJDK替换java API

5.Android7.0采用了一项具有实时代码剖析功能的ARI JIT编译器,它能够在安卓应用程序在运行时不断提高自身的性能

Android 7.0 引入一项新的应用签名方案 APK Signature Scheme v2，它能提供更快的应用安装时间和更多针对未授权 APK 文件更改的保护。在默认情况下，Android Studio 2.2 和 Android Plugin for Gradle 2.2 会使用 APK Signature Scheme v2 和传统签名方案来签署您的应用。

这是否意味着v2签名的apk只能在7.0手机中安装，这有待验证。来自百度的一些答案：

问题描述(v1和v2)

Android 7.0中引入了APK Signature Scheme v2，v1呢是jar Signature来自JDK

V1：应该是通过ZIP条目进行验证，这样APK 签署后可进行许多修改 - 可以移动甚至重新压缩文件。

V2：验证压缩文件的所有字节，而不是单个 ZIP 条目，因此，在签名后无法再更改(包括 zipalign)。正因如此，现在在编译过程中，我们将压缩、调整和签署合并成一步完成。好处显而易见，更安全而且新的签名可缩短在设备上进行验证的时间（不需要费时地解压缩然后验证），从而加快应用安装速度。

解决方案一

v1和v2的签名使用

只勾选v1签名并不会影响什么，但是在7.0上不会使用更安全的验证方式

只勾选V2签名7.0以下会直接安装完显示未安装，7.0以上则使用了V2的方式验证

同时勾选V1和V2则所有机型都没问题

解决方案二

在app的build.gradle的android标签下加入如下

signingConfigs {

    debug {

        v1SigningEnabled true

        v2SigningEnabled true

    }

    release {

        v1SigningEnabled true

        v2SigningEnabled true

    }

}

个人建议：

这是一个从Android7.0系统出现的新的签名机制，这个新机制使得apk的签名方式更加安全，首先我们应该尝试把V1和V2两个选项全部勾选，并尽可能的去使用他们，但是如果全部勾选完毕出现了问题，那么我们可以忽略这种新的签名机制，只勾选第一个选项(V1)，依旧使用我们之前老的签名机制。

这有Android源文介绍：<https://source.android.com/security/apksigning/v2>

Android Signature V2 Scheme签名下的新一代渠道包打包神器：http://www.ctolib.com/walle.html，这是美团推出的打包工具：<http://tech.meituan.com/android-apk-v2-signature-scheme.html>

另一遍相关文章：<http://www.jianshu.com/p/e1e2fd05bb62>