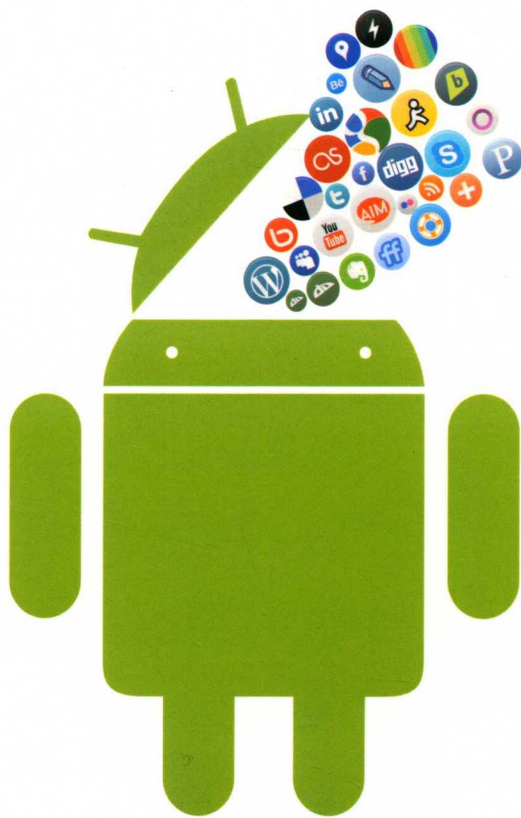


从规划到上线全程详解

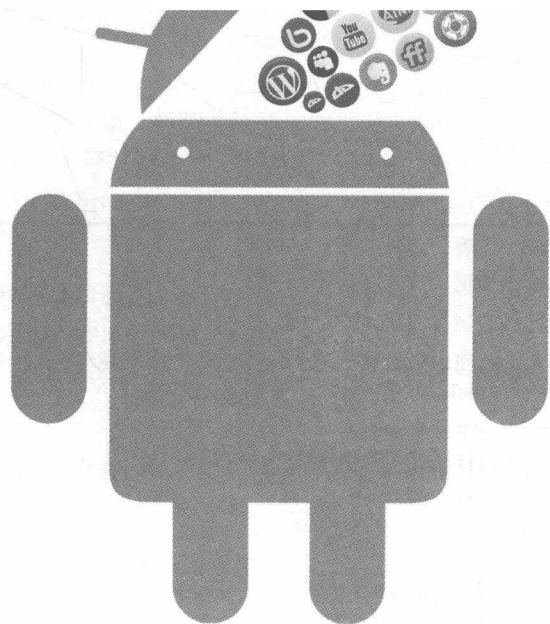
# Android APP 开发实战

强增 编著



# Android APP 开发实战

强增 编著



人民邮电出版社  
北京

## 图书在版编目 (C I P) 数据

Android APP开发实战：从规划到上线全程详解 /  
强增编著. — 北京：人民邮电出版社，2018.6  
ISBN 978-7-115-47230-4

I. ①A… II. ①强… III. ①移动终端—应用程序—  
程序设计 IV. ①TN929.53

中国版本图书馆CIP数据核字(2017)第278625号

## 内 容 提 要

本书详尽地介绍了 APP 从规划到上线过程中所涉及的各方面知识，包括产品规划和原型设计、效果图设计、切图和尺寸标注、APP 与服务器间的协作、字符编码、网络、多媒体、数据加密、设计模式、APP 架构设计、APP 功能开发、APP 性能优化、开发工具的使用、APP 测试和项目管理等。本书帮助读者开阔眼界，且能够以更专业和高效的方式完成 APP 的开发，适合新 APP 工程师作为开发参考。

---

◆ 编 著 强 增

责任编辑 赵 轩

责任印制 焦志炜

◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号

邮编 100164 电子邮件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

三河市祥达印刷包装有限公司印刷

◆ 开本：800×1000 1/16

印张：20

字数：516 千字

2018 年 6 月第 1 版

印数：1—2 400 册

2018 年 6 月河北第 1 次印刷

---

定价：69.00 元

读者服务热线：(010)81055410 印装质量热线：(010)81055316

反盗版热线：(010)81055315

广告经营许可证：京东工商广登字 20170147 号

随着移动互联网的兴起，各行各业对移动应用的需求越来越大，从事 APP 开发的人也越来越多，APP 开发行业可以说是方兴未艾。APP 开发是比较复杂的事情，涉及产品、美工设计、服务器端开发、Android/iOS 开发、软件测试和项目管理等各方面。这些方面都是相互关联的，如果要做好一个 APP，需要对上述各方面都有所了解。

目前介绍 APP 开发的书籍很多，大都局限于某一方面，内容不够全面，并且许多书籍偏于理论，与实际联系不紧密。国内外市面上大多数的 Android 开发类图书，基本上可以分为两类：一类是从系统内核和源代码入手，书的内容重在分析 Android 各个模块的运行机制，深入理解系统肯定对应用开发者有好处，但很多时候并不是那么实用；另一类是标准教程，基本内容与 Android 官方文档类似，围绕 API 的用法就事论事地讲解，这类书在写法、教学思路和实例上虽然各有千秋，但在实际工作中就会发现还远远不够。

在实际工作中，笔者接触了许多从别的行业转行从事 APP 开发或从事 APP 开发一两年的人员，发现许多人对 APP 开发的基础知识不甚了解，需要关注的地方都没有考虑到，导致在开发 APP 的过程中犯了许多低级错误，而这些本来是可以避免的；而在 APP 开发行业中，从事 APP 开发一两年的人员又是占了绝大多数。在许多创业型的小公司里，也没有专职的产品、美工和测试人员，往往要求 Android/iOS 开发人员做到一专多能、身兼数职。

有感于此，笔者想写一本面向初级读者且全面介绍 APP 开发的书。与上述两类书都不同，本书完全从实战出发，以从零开始规划一款 APP 至 APP 上线这个过程为主线，介绍整个过程中所涉及的知识，而不局限于某一方面。这样一来，读者将对 APP 开发的相关知识有一个全面的了解，进而以更专业的方式完成 APP 的开发。

本书是从多个实际项目中获得的经验总结，可以使读者在开发 APP 的过程中少犯低级错误，减少可能遇到的各种问题，从而提高开发效率。

本书的章节编排贴合实际项目，具体内容如下所述。

- 第 1 章：介绍开发 APP 前的准备事项。
- 第 2 章：介绍 APP 产品和设计人员的工作（产品原型设计、效果图设计和切图等）。
- 第 3 章：讲解 APP 端和服务器端的协作（接口设计、数据安全方案、数据同步方案和登录状态

的维持)。

- 第4章~第9章：讲解字符编码、网络（TCP/IP、Socket、HTTP/HTTPS、Cookie 和 Session）、数据加密和设计模式等知识。
- 第10章~第11章：介绍 APP 架构设计。
- 第12章~第24章：详解 Android 开发（四大组件、Fragment、权限功能、动画实现、矢量图的使用、异常处理、本地存储、ABI 管理和混淆功能）。
- 第25章~第32章：涵盖 Android Studio 的使用技巧、APP 的缓存设计、APP 的性能优化、APP 的 Log 功能设计、APP 的版本管理、APP 版本升级功能设计、APP 常用功能设计和代码封装。
- 第33章：介绍 APP 测试。
- 第34章：介绍 APP 项目管理。
- 第35章：讲解 Git 的使用。

## 资源与支持

本书由异步社区出品，社区（<https://www.epubit.com/>）为您提供相关资源和后续服务。

### 配套资源

---

本书提供如下资源：

- 本书配套资源请到异步社区的本书购买页面中下载。

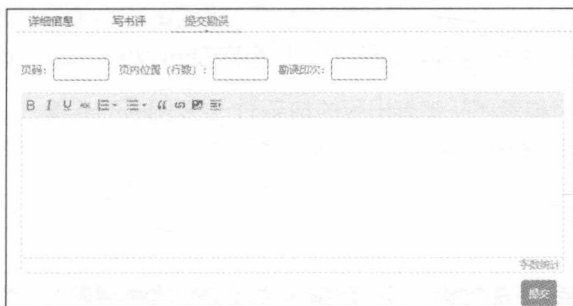
要获得以上配套资源，请在异步社区本书页面中点击 **配套资源**，跳转到下载界面，按提示进行操作即可。注意：为保证购书读者的权益，该操作会给出相关提示，要求输入提取码进行验证。

### 提交勘误

---

作者和编辑尽最大努力来确保书中内容的准确性，但难免还会存在疏漏。欢迎您将发现的问题反馈给我们，帮助我们提升图书的质量。

当您发现错误时，请登录异步社区，搜索到本书页面，点击“提交勘误”，输入相关信息，单击“提交”按钮即可。本书的作者和编辑会对您提交的勘误进行审核，确认并接受后，您将获赠异步社区的100积分。积分可用于在异步社区兑换优惠券，或者用于兑换样书或奖品。



## 扫码关注本书

扫描下方二维码，您将会在异步社区微信服务号中看到本书信息及相关的服务提示。



## 与我们联系

我们的联系邮箱是 [contact@epubit.com.cn](mailto:contact@epubit.com.cn)。

如果您对本书有任何疑问或建议，请您发邮件给我们，并在邮件标题中注明本书书名，以便我们更高效地做出反馈。

如果您有兴趣出版图书、录制教学视频，或者参与图书翻译、技术审校等工作，可以发邮件给我们，或者到异步社区在线提交投稿（直接访问 [www.epubit.com/selfpublish/submission](http://www.epubit.com/selfpublish/submission) 即可）。

如果您是学校、培训机构或企业，想批量购买本书或异步社区出版的其他图书，也可以发邮件给我们。

如果您在网上发现有针对异步社区出品图书的各种形式的盗版行为，包括对图书全部或部分内容的非授权传播，请您将怀疑有侵权行为的链接发邮件给我们。您的这一举动是对作者权利的保护，也是我们持续为您提供有价值的内容的动力之源。

## 关于异步社区和异步图书

“异步社区”是人民邮电出版社旗下 IT 专业图书社区，致力于出版精品 IT 技术图书和相关学习产品，

为译者提供优质出版服务。社区创办于 2015 年 8 月，提供超过 1000 种图书、近千种电子书，以及众多技术文章和视频课程。更多详情请访问异步社区官网 <https://www.epubit.com>。

“异步图书”是由异步社区编辑团队策划出版的精品 IT 专业图书的品牌，依托于人民邮电出版社近 30 年的计算机图书出版积累和专业编辑团队，相关图书在封面上印有异步图书的 LOGO。异步图书的出版领域包括软件开发、大数据、AI、测试、前端、网络技术等。



<b>第1章 开发APP前的准备事项</b> .....	<b>1</b>	<b>3.3 登录方式</b> .....	<b>32</b>
1.1 APP适配的硬件 .....	2	3.3.1 账号+密码 .....	32
1.2 APP横竖屏界面的选择 .....	2	3.3.2 账号+密码+验证码 .....	32
1.3 APP适配的操作系统版本 .....	2	<b>3.4 登录状态的维持</b> .....	<b>33</b>
1.4 APP适配的屏幕尺寸 .....	3	3.4.1 利用Token实现 .....	33
1.5 APP开发样机的选择 .....	4	3.4.2 利用Cookie实现 .....	34
1.6 APP内置的图片资源 .....	4	3.4.3 利用账号和密码实现 .....	35
<b>第2章 APP产品和设计人员的工作</b> ...	<b>5</b>	<b>3.5 数据同步方案</b> .....	<b>35</b>
2.1 编写产品文档时的注意事项 .....	6	3.5.1 文件的同步 .....	35
2.2 设计效果图时的注意事项 .....	17	3.5.2 地址数据的同步 .....	36
2.3 APP的切图工作 .....	18	3.5.3 非地址数据的同步 .....	37
2.4 点9图简介 .....	20	<b>3.6 业务逻辑的实现</b> .....	<b>38</b>
2.4.1 点9图 .....	20	<b>3.7 接口文档的维护</b> .....	<b>38</b>
2.4.2 制作工具 .....	20	<b>第4章 字符编码</b> .....	<b>40</b>
2.4.3 注意事项 .....	21	4.1 字符集 .....	41
2.5 产品和设计文档的版本管理 .....	21	4.2 字符编码 .....	42
<b>第3章 APP端和服务器端的协作</b> .....	<b>22</b>	4.3 字节序 .....	43
3.1 接口设计注意事项 .....	23	<b>第5章 TCP/IP概述</b> .....	<b>44</b>
3.2 安全方案 .....	30	5.1 协议简介 .....	45
3.2.1 HTTP命令的使用 .....	30	5.2 TCP和UDP的区别 .....	45
3.2.2 使用时间戳参数 .....	31	5.2.1 面向连接服务 .....	45
3.2.3 数据加密 .....	32	5.2.2 无连接服务 .....	46
3.2.4 密码的处理 .....	32		
3.2.5 数据的存储 .....	32		



<b>第6章 HTTP网络请求 .....</b>	<b>47</b>	<b>第9章 设计模式 .....</b>	<b>80</b>
6.1 HTTP简介 .....	48	9.1 设计模式简介 .....	81
6.1.1 协议 .....	48	9.2 面向对象设计原则 .....	81
6.1.2 HTTP方法 .....	48	9.3 设计模式类别 .....	81
6.1.3 HTTP消息 .....	49	9.3.1 单例模式 .....	82
6.1.4 HTTP头字段介绍 .....	52	9.3.2 Builder模式 .....	83
6.1.5 Keep-Alive模式介绍 .....	55	9.3.3 原型模式 .....	83
6.1.6 HTTP状态码简介 .....	56	9.3.4 工厂方法模式 .....	83
6.2 Cookie简介 .....	56	9.3.5 策略模式 .....	84
6.2.1 Cookie .....	56	9.3.6 状态模式 .....	84
6.2.2 Cookie的设置和发送 .....	57	9.3.7 命令模式 .....	85
6.3 Session简介 .....	57	9.3.8 观察者模式 .....	85
6.3.1 Session .....	57	9.3.9 备忘录模式 .....	85
6.3.2 SessionID .....	58	9.3.10 迭代器模式 .....	85
6.4 短连接与长连接 .....	58	9.3.11 模板方法模式 .....	85
6.4.1 短连接 .....	58	9.3.12 代理模式 .....	85
6.4.2 长连接 .....	58	9.3.13 组合模式 .....	86
6.4.3 使用场景 .....	59	9.3.14 适配器模式 .....	86
6.5 Volley网络库简介 .....	59	9.3.15 外观模式 .....	86
6.5.1 Volley网络库 .....	59	9.3.16 桥接模式 .....	86
6.5.2 Volley网络库的使用 .....	66	<b>第10章 架构模式 .....</b>	<b>87</b>
<b>第7章 HTTPS概述 .....</b>	<b>73</b>	10.1 MVC架构 .....	88
7.1 协议简介 .....	74	10.2 MVP架构 .....	88
7.2 HTTPS的认证类型 .....	74	10.3 MVVM架构 .....	89
7.2.1 单向认证 .....	74	10.4 MVP+VM架构 .....	89
7.2.2 双向认证 .....	75	<b>第11章 APP架构设计 .....</b>	<b>90</b>
<b>第8章 加密简介 .....</b>	<b>76</b>	11.1 基本原则 .....	91
8.1 对称加密 .....	77	11.2 分层设计 .....	92
8.2 非对称加密 .....	77	11.2.1 三层架构 .....	92
8.3 MD5简介 .....	77	11.2.2 View层设计 .....	92

11.2.3 业务逻辑层设计 (Presenter) .....	94	13.3 IntentService .....	116
11.2.4 数据访问层设计 (Model) ..	94	13.4 前台服务 .....	116
11.2.5 功能模块设计 .....	94	13.5 服务的生命周期 .....	117
11.3 层间通信 .....	96	<b>第14章 Broadcast概述 .....</b>	<b>118</b>
11.3.1 通信方式 .....	96	14.1 广播机制简介 .....	119
11.3.2 交互模式 .....	96	14.2 BroadcastReceiver .....	119
11.4 跨业务模块调用 .....	97	14.2.1 静态注册 .....	119
11.4.1 跨业务模块调用简介 .....	97	14.2.2 动态注册 .....	120
11.4.2 跨业务模块调用方案 .....	97	14.3 广播类型 .....	120
<b>第12章 Activity概述 .....</b>	<b>98</b>	14.3.1 普通广播 (Normal Broadcast) ..	120
12.1 Activity启动方式 .....	99	14.3.2 系统广播 (System Broadcast) ..	121
12.1.1 启动模式 .....	99	14.3.3 有序广播 (Ordered Broadcast) ..	121
12.1.2 FLAG介绍 .....	100	14.3.4 局部广播 (Local Broadcast) ..	123
12.2 Activity消息路由 .....	101	14.4 广播的安全性 .....	123
12.2.1 设计思路 .....	101	<b>第15章 ContentProvider概述 .....</b>	<b>125</b>
12.2.2 具体实现 .....	101	<b>第16章 Fragment概述 .....</b>	<b>127</b>
12.3 Activity数据的保存和恢复 ..	103	16.1 Fragment简介 .....	128
12.3.1 临时保存数据和恢复数据 ..	103	16.2 Fragment的创建 .....	128
12.3.2 持久保存数据和恢复数据 ..	103	16.3 Fragment的懒加载 .....	130
12.4 Activity数据传递 .....	104	16.4 Fragment的数据保存和恢复 ..	133
12.4.1 数据传递媒介 .....	104	16.4.1 临时保存数据和恢复 .....	133
12.4.2 数据传递机制 .....	108	16.4.2 持久保存数据和恢复 .....	134
12.5 BaseActivity设计 .....	109	16.5 Fragment的使用场景 .....	134
12.5.1 应用级别的BaseActivity 设计 .....	109	<b>第17章 Android权限 .....</b>	<b>135</b>
12.5.2 功能级别的BaseActivity 设计 .....	111	17.1 权限分类 .....	136
<b>第13章 Service概述 .....</b>	<b>114</b>	17.1.1 Normal Permissions ..	136
13.1 Service的不同形式 .....	115		
13.2 Service与线程 .....	115		

17.1.2	Dangerous Permissions	137	21.2.3	ANR异常的处理	170
17.2	动态权限申请	138	21.3	注意事项	174
17.3	兼容性问题	139	<b>第22章 Android的本地存储 175</b>		
<b>第18章 Android动画 140</b>			22.1	内部存储 (Internal Storage)	176
18.1	帧动画	141	22.1.1	非缓存文件的处理	176
18.2	View动画	142	22.1.2	缓存文件的处理	176
18.3	属性动画简介	144	22.2	外部存储 (External Storage)	177
18.3.1	属性动画	144	22.2.1	外部公共存储	178
18.3.2	使用示例	145	22.2.2	外部私有存储	179
18.4	Activity切换动画	147	22.2.3	使用作用域目录访问	182
<b>第19章 图片类型 149</b>			<b>第23章 ABI管理 184</b>		
19.1	位图简介	150	23.1	ABI简介	185
19.1.1	位图	150	23.2	支持的ABI	185
19.1.2	WebP格式	150	23.3	为特定ABI生成代码	186
19.2	矢量图简介	151	23.4	Android系统的ABI管理	186
<b>第20章 Android矢量图的使用 152</b>			23.5	Android系统ABI支持	187
20.1	功能简介	153	23.6	安装时自动解压缩原生代码	187
20.2	兼容性处理	154	<b>第24章 ProGuard混淆 188</b>		
20.3	Vector语法简介	155	24.1	ProGuard简介	189
20.4	Vector静态图的使用	157	24.2	ProGuard指令介绍	189
20.5	Vector动态图的使用	159	24.3	ProGuard注意事项	190
20.5.1	功能实现	159	24.4	ProGuard相关文件	192
20.5.2	动态Vector兼容性问题	161	<b>第25章 Android Studio使用技巧 193</b>		
<b>第21章 Android异常 162</b>			25.1	编译打包	194
21.1	异常分类	163	25.2	功能宏的使用	196
21.2	异常处理	163	25.3	集成SO文件	196
21.2.1	使用try...catch...处理异常	164	25.4	模板的定制使用	197
21.2.2	使用UncaughtExceptionHandler处理异常	164	<b>第26章 APP缓存处理 211</b>		
			26.1	缓存简介	212

26.2	缓存控制 .....	212	28.4.1	AOP简介 .....	239
26.3	缓存实现 .....	212	28.4.2	AOP技术的使用 .....	240
26.4	WebView缓存 .....	213	<b>第29章</b>	<b>APP版本管理 .....</b>	<b>254</b>
26.5	缓存注意事项 .....	214	<b>第30章</b>	<b>APP版本更新功能设计 .....</b>	<b>256</b>
26.6	清除数据和清除缓存的区别 ..	214	30.1	功能项 .....	257
<b>第27章</b>	<b>APP性能优化 .....</b>	<b>215</b>	30.1.1	服务器端功能 .....	257
27.1	减少APP所占空间大小 .....	216	30.1.2	APP端功能 .....	257
27.1.1	减少图片所占空间大小 ..	216	30.2	APP和服务器交互 .....	257
27.1.2	减少音频文件所占空间 大小 .....	221	<b>第31章</b>	<b>APP常用功能设计 .....</b>	<b>260</b>
27.1.3	减少代码所占空间大小 ..	221	31.1	启动界面设计 .....	261
27.1.4	使用APK Analyzer分析 APP .....	222	31.1.1	启动界面白屏解决方案 .....	261
27.1.5	利用工具减少APP大小 ..	226	31.1.2	启动界面屏蔽返回按键 .....	261
27.2	减少APP使用的网络流量 .....	228	31.2	首页设计 .....	262
27.3	内存优化 .....	229	31.3	登录功能设计 .....	262
27.3.1	节省内存 .....	229	31.4	商品详情界面设计 .....	262
27.3.2	防止内存泄露 .....	231	31.5	购物车功能设计 .....	262
27.3.3	防止OOM .....	232	31.6	商品展示界面功能设计 .....	263
27.4	UI性能优化 .....	232	31.7	个人中心界面功能设计 .....	263
27.5	电量优化 .....	233	31.8	搜索功能设计 .....	263
27.6	运行速度优化 .....	233	31.9	WebView功能设计 .....	264
27.7	性能优化工具 .....	234	31.10	出错提示功能设计 .....	266
27.7.1	Android Studio自带 工具 .....	234	31.11	界面内容隐藏和显示设计 .....	266
27.7.2	Android系统工具 .....	235	31.12	提示功能设计 .....	267
27.7.3	三方工具 .....	236	31.12.1	三种控件简介 .....	267
<b>第28章</b>	<b>Log功能设计 .....</b>	<b>237</b>	31.12.2	AlertDialog介绍 .....	268
28.1	Log输出控制 .....	238	31.12.3	Toast介绍 .....	269
28.2	注意事项 .....	239	31.12.4	Snackbar介绍 .....	270
28.3	Log数据的格式化 .....	239	31.13	定期执行任务的功能设计 .....	271
28.4	使用AOP技术输出Log .....	239	31.13.1	JobScheduler介绍 .....	271
			31.13.2	JobScheduler的替代 方案 .....	274

31.13.3 注意事项 .....	277	31.18.2 使用MockWebServer进行 接口的单元测试 .....	286
<b>31.14 全屏模式的功能设计 .....</b>	<b>277</b>	<b>31.19 自动调整文字大小的 TextView .....</b>	<b>289</b>
31.14.1 Lean Back .....	277	31.19.1 Default方式 .....	290
31.14.2 Immersive .....	278	31.19.2 Granularity方式 .....	290
<b>31.15 开机自启动的功能设计 .....</b>	<b>279</b>	31.19.3 Preset Sizes方式 .....	291
31.15.1 普通模式 .....	279	<b>第32章 代码封装 .....</b>	<b>293</b>
31.15.2 直接启动模式 .....	279	<b>第33章 APP测试 .....</b>	<b>295</b>
31.15.3 示例代码 .....	279	<b>第34章 项目管理 .....</b>	<b>298</b>
<b>31.16 APP快捷图标的功能设计 .....</b>	<b>280</b>	34.1 项目团队成员 .....	299
31.16.1 简介 .....	280	34.2 需求处理 .....	299
31.16.2 静态快捷图标 .....	281	34.3 进度计划 .....	300
31.16.3 动态快捷图标 .....	282	<b>第35章 Git使用 .....</b>	<b>302</b>
<b>31.17 针对Android7.0及更高版本的 后台优化方案 .....</b>	<b>282</b>	35.1 Git工具简介 .....	303
31.17.1 对于CONNECTIVITY_ ACTION限制的解决 方案 .....	283	35.1.1 客户端工具 .....	303
31.17.2 对于ACTION_NEW_ PICTURE和ACTION_ NEW_VIDEO限制的 解决方案 .....	284	35.1.2 服务器端工具 .....	306
<b>31.18 服务器接口的单元测试 .....</b>	<b>285</b>	35.2 Git常用命令 .....	306
31.18.1 单元测试 .....	286	35.3 使用Git的注意事项 .....	307

# 第1章 开发APP前的准备事项

- 1.1 APP 适配的硬件
- 1.2 APP 横竖屏界面的选择
- 1.3 APP 适配的操作系统版本
- 1.4 APP 适配的屏幕尺寸
- 1.5 APP 开发样机的选择
- 1.6 APP 内置的图片资源

开发 APP 前，除了竞品分析和项目组搭建外，还有以下事项需要确定。

## 1.1 APP适配的硬件

大多数用户主要是将 PAD 用于娱乐、教育或企业办公。对于游戏、视频播放、图形图像处理、阅读、教育或企业办公等类型的 APP，建议适配手机和 PAD。电商或理财等类型的 APP 建议只适配手机。

随着硬件性能和网速的提高，使用 PAD 的用户越来越倾向于直接打开相关的网站，而不愿去下载应用，以免频繁升级应用。因此在开发 APP 的时候，为节约资源，加快开发进度，可以先开发网站和手机版本的 APP，最后再开发 PAD 版本的 APP。

## 1.2 APP横竖屏界面的选择

目前大部分 PAD 的尺寸都是大于等于 7.9 英寸，对于 PAD 版本的应用可以只考虑开发横屏界面，不用考虑竖屏界面。还有一部分 PAD 的尺寸在 7 英寸左右，对于这类 PAD，可以考虑直接让用户使用手机版本的 APP（如果一个 APP 在 6 英寸的手机上使用，没有界面问题，那在 7 英寸的 PAD 上通常也可以正常使用）。

对于手机版本的应用：游戏、视频播放、图形图像处理、阅读、教育或企业办公等类型的 APP 建议支持横屏和竖屏；电商或理财等类型的 APP 可以只支持竖屏。

## 1.3 APP适配的操作系统版本

iOS APP 建议适配 iOS 8.0 以上版本，Android APP 建议适配 Android 4.2 以上版本，可以根据 Apple 和 Google 提供的各版本系统占有率的统计数据做实时调整。使用 Android Studio 新建工程时，在图 1-1 所示的界面，单击“Help me choose”链接，显示如图 1-2 所示的界面，列出了 Android 系统各版本的市场占有率。

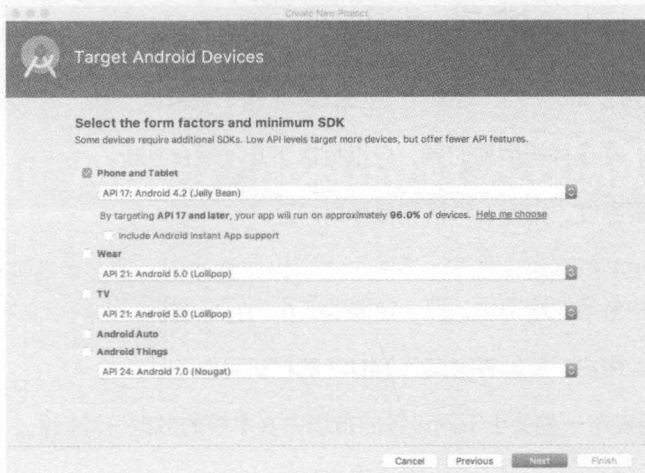


图1-1

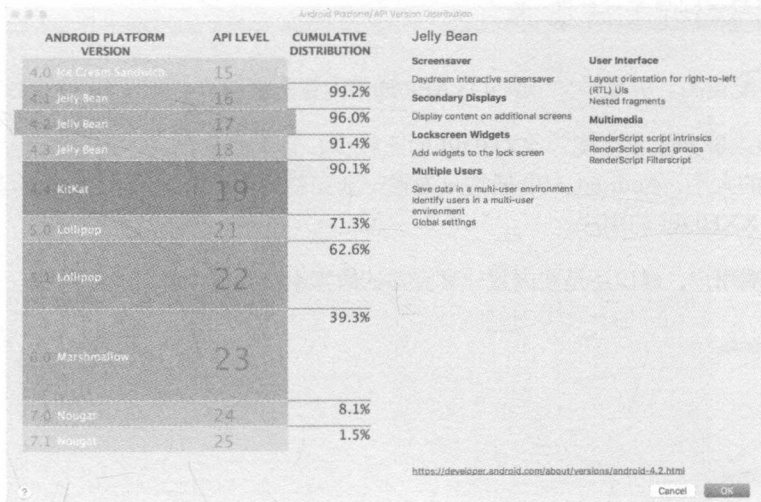


图1-2

对于大多数公司来说，毕竟开发和测试资源有限，对于占有率低于5%的系统版本建议不必考虑支持。

## 1.4 APP适配的屏幕尺寸

Apple 产品的尺寸不像 Android 产品那么多，每种尺寸的用户量巨大，除了 iPhone 4 系列和之前系列的手机外，其余型号的手机和 PAD 都要适配。

Android 产品的尺寸千差万别，建议适配的手机屏幕基准尺寸为 5 英寸、5.5 英寸和 6 英寸，PAD 屏幕基准尺寸为 7 英寸、8 英寸和 10 英寸；屏幕像素密度支持 MDPI、HDPI、XHDPI 和 XXHDPI 这 4 种。



## 1.5 APP开发样机的选择

Apple 的机型比较少，除了 iPhone 4 系列的手机外，其余每个机型都可以考虑购买样机。

Android 的机型比较多，在选择机型的时候，应根据以下几项综合判断。

- 屏幕尺寸、屏幕像素密度、内存、价格和目标用户。
- APP 在小内存（1GB）低端机（低于 1 千元）上可以正常使用，那在中高端机上肯定也可以正常使用，如果目标用户包括低端用户，那一定要购买小内存低端机，大内存高端机可以不买。
- 如果不考虑低端设备用户，那屏幕密度为 MDPI 的样机可以不买。

友盟和极光等厂商也会发布一些关于不同设备的市场占有率等的数据统计报告，在选择开发样机的时候，可以做参考。

## 1.6 APP内置的图片资源

Apple 的产品硬件配置高、屏幕好，所以 iOS APP 通常内置两套图片，即 @2x 和 @3x 分辨率的图片。

Android 的产品中，屏幕像素密度为 XXHDPI 的比较少，且大多数产品的配置不高，为了节约存储空间和减少 APP 的大小，Android APP 通常只内置一套完整的像素密度为 XHDPI 的图片，以及部分 MDPI、HDPI 和 XXHDPI 的图片。

如果 APP 定位高端用户，建议还是要内置一套完整的像素密度为 XXHDPI 的图片。