

移动App测试的 22条军规

Pro Mobile App Testing With
22 Rules

黄勇 著

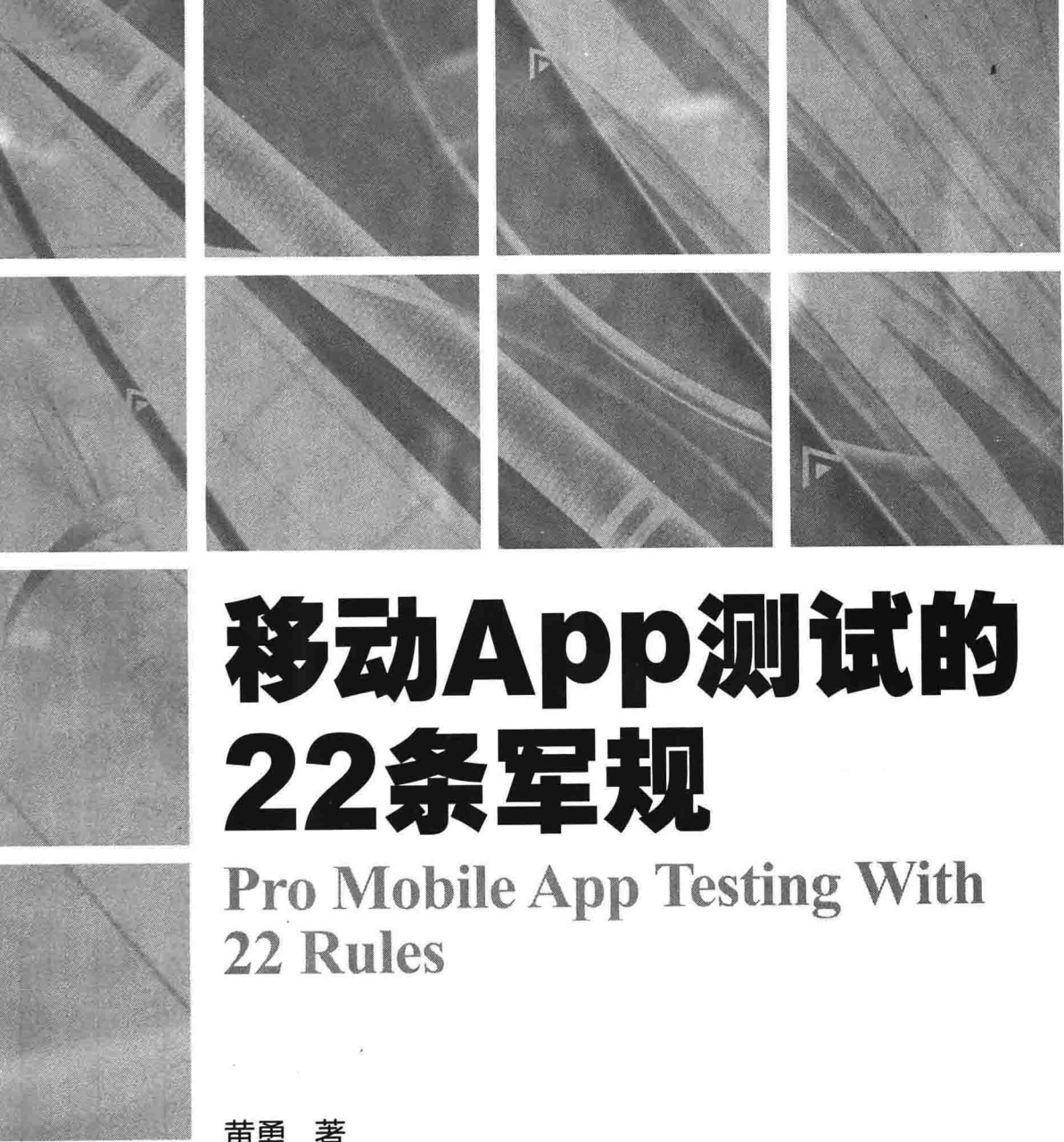
- 详细讲述了移动App测试中必知必会的技术、工具和技巧，以及测试用例、方法和案例
- 通过22条来自于测试实战中的最佳实践，帮助读者全面理解移动App测试的中心思想，在复杂的“无规则”中找到规则并进行高效测试
- 图文并茂，帮助读者更快速掌握移动App测试全流程



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS



移动App测试的 22条军规

Pro Mobile App Testing With
22 Rules

黄勇 著

人民邮电出版社
北京

图书在版编目 (C I P) 数据

移动App测试的22条军规 / 黄勇著. — 北京 : 人民
邮电出版社, 2015. 7
ISBN 978-7-115-39496-5

I. ①移… II. ①黄… III. ①移动终端—应用程序—
程序测试—指南 IV. ①TN929.53-62

中国版本图书馆CIP数据核字(2015)第137332号

内 容 提 要

本书全面讲解了移动 App 测试的技术、技巧、工具、案例和测试用例, 全书共分 23 章, 主要内容为: 移动 App 的特性, 关注多任务和意外情况处理, 避免手势冲突, 关注用户体验, 设计通知和消息展示, 支持操作系统特性, 及时显示和同步消息, 支持多种文件格式, 支持多语言和地区设置, 重点测试高内存占用的功能、降低流量和电量消耗, 确保成功集成和调用第三方 App, 尽量不使用非标准控件, iOS 8 升级所引入的新特性, Android 5.0 升级所引入的新特性, 自动化和探索性测试, 自动化测试中模拟器的使用, 用户界面自动化测试的常见工具, 性能和安全性测试, 使用 Log 定位问题, 充分使用持续集成、持续部署, 以及微信 App 测试综合案例分析等核心技术。

本书适合软件的测试初学者、测试从业人员及程序员阅读, 也可以作为大专院校相关专业师生的学习用书, 以及培训学校的教材。

-
- ◆ 著 黄 勇
责任编辑 张 涛
责任印制 张佳莹 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>
北京艺辉印刷有限公司印刷
 - ◆ 开本: 800×1000 1/16
印张: 16.25
字数: 326 千字
印数: 1—3 000 册
- 2015 年 7 月第 1 版
2015 年 7 月北京第 1 次印刷
-

定价: 49.00 元

读者服务热线: (010)81055410 印装质量热线: (010)81055316
反盗版热线: (010)81055315

自序

“移动 App 测试”，哦？虽然我们开发移动 App，但我又不是测试人员，关心测试干什么？

先别急着打退堂鼓，试想一下在移动 App 开发团队中谁需要了解用户？谁需要知道技术实现？再想一下对这两方面都了解的有谁？难道不应该是测试人员吗？

现在你还不想了解测试人员是如何看待和关注移动 App，以及在工作中是如何融合用户需求和技术的实现吗？

本书中这 22 条军规不仅针对测试人员，对于开发人员和项目经理同样适用。我们在开发移动 App 的时候，无论是从用户角度进行需求分析，还是从技术实现角度构建 App，都可以遵循这 22 条军规的指导。

本书介绍的 App 测试军规绝没有枯燥的理论，都是实践案例的浓缩，在实例的介绍中没准就有你的项目的影子，为什么不按照军规实战一把呢？

希望通过笔者的经验分享能让你少走一些弯路。

前 言

随着这几年智能设备的大规模普及，越来越多的人开始使用智能手机和平板电脑，因此，移动 App 的使用也越来越广泛。用户对于手机和平板电脑上 App 的要求渐渐地不局限于功能，更多地要求提升用户体验，且蜂拥进入移动 App 市场的各类公司也在加剧着这种趋势。因此，要想使自己的 App 脱颖而出，不仅需要使产品以更高品质，在更短的时间内投放到市场，还需要不断改进产品，以满足用户不断变化的需求和体验。

对整个 App 开发团队来说，这是很大的挑战，因为，不仅需要从开发方法和流程上适应各种变化，还需要更新技能以适应这些变化。对测试人员来说，了解移动 App 的测试和桌面应用测试的区别，设计专门针对移动 App 的测试场景和用例，高效地进行移动 App 的测试，成为了头等大事，为了帮助读者尽快适应这样的需求，特意撰写了本书。

本书是作者从项目实践中总结出来的 22 条移动 App 测试军规，帮助读者梳理测试思维，指导读者设计测试用例，以便更好地完成 App 的完整测试，本书的主要内容如下。

移动 App 的特性，关注多任务和意外情况处理，避免手势冲突，关注用户体验，其他需要关注的用户体验的细节，设计通知和消息展示，支持操作系统特性，及时显示和同步消息，支持多种文件格式，支持多语言和地区设置，重点测试高内存占用的功能，降低流量和电量消耗，确保成功集成和调用第三方 App，测试 App 使用社交媒体等账号登录的功能，iOS 8 升级所引入的新特性，Android 5.0 升级所引入的新特性，尽量减少依赖，进行自动化和探索性测试，自动化测试中模拟器的使用，用户界面测试，性能和安全性测试，测试 App 用到的后台服务 Mobile Service 的性能，使用 Log 定位问题，充分使用持续集成和持续部署，微信 App 测试综合案例分析。

由于写作时间仓促，加之作者水平有限，书中难免有不当的地方，恳请广大读者给出修改建议，本书答疑联系方式为：<http://weibo.com/hy1984427>。编辑联系邮箱为：zhangtao@ptpress.com.cn。

本书可以作为测试初学者阅读，可以帮助读者快速融入测试行业，并全面了解和掌握 App 测试所需要的技术和方法；本书适合测试从业人员阅读，通过本书讲述的技

术、技巧、工具、案例和测试用例，可以帮助读者尽快进行自己项目的测试，因为本书所讲的技术适用于任何移动 App 测试项目；本书也适合从业于移动 App 开发的程序员，可以从本书了解 App 测试在整个产品开发中的位置和重要性，并在工作中与测试人员紧密配合，高效完成测试的全过程；同时本书也适合大专院校相关专业师生的学习用书，以及培训学校的教材。

目 录

军规 1 确定设备和平台再动手	1
1.1 移动 App 的特性	1
1.2 移动 App 的生命周期	3
1.3 设备的硬件参数	6
军规 2 “移动”测试	11
军规 3 关注多任务和意外情况处理	21
3.1 第一个场景	22
3.2 第二个场景	23
3.3 需注意的场景	24
3.4 硬件的影响	25
军规 4 避免手势冲突	27
4.1 从屏幕左侧边缘向右滑动	27
4.2 在屏幕上向左滑动	31
4.3 从屏幕顶部向下滑动	32
4.4 从屏幕底部向上滑动	33
4.5 按住屏幕向下滑动	34
4.6 在图片上双击	35
4.7 两根手指分开和捏合	36
4.8 两根手指按住屏幕旋转	37
4.9 3 根手指的手势操作	37
4.10 4 根手指向上/下滑动	38
4.11 4 根手指向左/右滑动	39
4.12 5 根手指聚拢的捏合操作	40
4.13 摇动设备	40

4.14	长按屏幕	41
军规 5	关注用户体验	43
5.1	横竖屏幕测试	44
5.2	WebView 的测试	46
5.3	规范与习惯	50
5.4	关注用户体验	51
5.5	其他需要关注的用户体验的小细节	54
军规 6	设计通知和消息展示	56
6.1	测试 App 安装时是否明确申明在用户使用 App 时需要用到的权限	56
6.2	测试 App 在用户使用过程中是否有合适的通知和消息显示	57
6.3	测试 App 在后台运行时是否有合适的通知和消息显示	62
6.4	测试 App 的消息推送功能	66
6.5	测试 App 在出错时是否有合适的通知和消息显示	67
军规 7	支持操作系统特性	70
7.1	Android App 测试设备的碎片化	70
7.2	Android App 更容易受到恶意软件的攻击	71
7.3	iOS 和 Android 对于 App 间通信的处理方式不一样	73
7.4	Android 和 iOS 就是否支持扩展存储有所不同	74
7.5	iOS 和 Android 对 Widget 的实现和使用不同	75
7.6	测试 Android App 对于 Dalvik 和 ART 运行环境 (RunTime) 的兼容性	78
7.7	测试 iOS App 在特定设置下的行为	79
军规 8	及时显示和同步消息	81
军规 9	适应特定用户界面 对功能和显示的影响	87
9.1	三星的 TouchWiz 用户界面	88
9.2	HTC 的 Sense 用户界面	89
9.3	LG 的 UX 用户界面	91
9.4	小米的米柚 MIUI 用户界面	92
9.5	魅族的 Flyme 用户界面	94

9.6	Sony 的 Xperia UI 用户界面	98
9.7	iOS App 的显式效果测试	98
军规 10	支持多种文件格式	100
10.1	App 支持 Office 文件	102
10.2	App 支持图片文件	102
10.3	App 支持视频和音频文件	103
军规 11	支持多语言和地区设置	105
11.1	App 不支持多语言和地区设置影响用户输入	106
11.2	App 不支持多语言和地区设置的影响	106
军规 12	重点测试高内存占用的功能	109
12.1	iOS 操作系统的内存管理机制以及对 App 使用内存的限制是很不透明的	110
12.2	Android 操作系统的内存管理机制更加透明，对 App 使用内存的限制也更加灵活	114
军规 13	降低流量和电量消耗	118
13.1	测试 App 安装文件的大小和安装过程	118
13.2	测试 App 占用的存储空间	119
13.3	测试 App 的流量消耗	120
13.4	测试 App 对于设备电量的消耗	122
军规 14	增量升级必不可少	124
14.1	测试 App 的增量升级	124
14.2	测试 App 的删除	127
14.3	测试 App 数据的清除	127
军规 15	确保成功集成和调用第三方 App	129
15.1	App 对第三方 App 的直接集成	130
15.2	测试 App 的分享功能	130
15.3	测试 App 显示外部链接的功能	131
15.4	测试免费 App 中集成广告的功能	132

15.5	测试 App 使用社交媒体等账号登录的功能	132
15.6	测试 App 推送服务	133
15.7	测试 App 关联其他文件的功能	134
15.8	测试 App 和输入法等 App 交互的功能	134
军规 16	尽量不使用非标准控件	136
军规 17	提前关注操作系统升级	140
17.1	iOS 6 升级所引入的新特性	141
17.2	iOS 7 升级所引入的新特性	142
17.3	iOS 8 升级所引入的新特性	144
17.4	Android 4.1 升级所引入的新特性	146
17.5	Android 4.4 升级所引入的新特性	148
17.6	Android 5.0 升级所引入的新特性	149
军规 18	尽量减少依赖	153
18.1	对于既有 Web 版本又有 App 版本的 App 要减少依赖	153
18.2	没有 Web 版本的 App 也需要考虑 App 的依赖	155
军规 19	进行自动化和探索性测试	167
19.1	测试设计和测试金字塔	167
19.2	单元和组件测试以及 TDD	170
19.3	Mobile Service 的 API 测试	171
19.4	用户界面的自动化测试	171
19.5	行为驱动开发 BDD	173
19.6	页面模式 Page Object	176
19.7	自动化测试中模拟器的使用	177
19.8	用户界面自动化测试的常见工具	179
19.9	探索性测试	182
军规 20	进行性能和安全性测试	186
20.1	测试 App 连接网络的速度	186
20.2	测试 App 在不同网络速度下操作的流畅程度	187

20.3	测试 App 对于前台页面渲染的性能	187
20.4	测试 App 操作数据库的性能	188
20.5	测试 App 用到的后台服务 Mobile Service 的性能	189
20.6	测试 App 是否保存了临时数据或者已删除的数据	194
20.7	测试 App 的会话 session 是否有过期设置	195
20.8	测试 App 请求中是否包含了明文的用户信息	195
20.9	测试 App 的请求是否加密	196
20.10	测试 SQLite 数据库的存储是否安全	197
20.11	测试 App 使用 WebView 的安全性	197
20.12	测试 App 的后台服务 Mobile Service	197
军规 21	使用 log 定位问题	202
军规 22	充分使用持续集成和持续部署	207
22.1	第一种方式	208
22.2	第二种方式	210
App 测试综合案例分析		211
23.1	首先需要确定测试微信 App 需要的设备和版本	211
23.2	“移动”测试微信 App	212
23.3	测试微信 App 的多任务和意外情况处理	214
23.4	测试微信 App 的手势操作	215
23.5	测试微信 App 的用户体验	216
23.6	测试微信 App 的消息显示和通知展示	220
23.7	测试微信 App 对于操作系统特性的支持程度	221
23.8	测试微信 App 能否及时显示和同步消息	221
23.9	测试微信 App 能否适应不同设备的不同用户界面	222
23.10	测试微信 App 对于多种格式图片的支持	223
23.11	测试微信 App 对多语言和地区的支持	224
23.12	测试微信 App 中高内存使用的功能	226
23.13	测试微信 App 的流量和电量消耗	227
23.14	测试微信 App 的增量升级	229
23.15	测试微信 App 中集成和调用第三方 App	231

23.16	测试微信 App 中非标准控件的使用情况	233
23.17	测试微信 App 对于最新操作系统特性的支持	234
23.18	测试微信 App 的依赖情况	235
23.19	对微信 App 进行自动化测试和探索性测试	235
23.20	对微信 App 进行性能测试和安全性测试	241
23.21	测试微信 App 的 log 提交	242
23.22	实现微信 App 的持续集成和持续部署	243
22 条军规之外	245
后记	246

军规 1 确定设备和平台再动手

在测试设计之初，测试人员首先会考虑的是什么呢？没错，就是测试的环境，也就是确定 App 究竟需要运行在什么样的设备和平台上。

显然，在移动设备和平台碎片化的现实中，测试人员穷尽所有设备和操作系统的版本来实现全覆盖的测试是不可能的。那如何在有限的时间和精力投入下，从投入产出比的角度出发，达到尽可能多的测试覆盖呢？这里主要考虑以下几个方面。

1.1 移动 App 的特性

(1) 如果 App 是针对心率监测、指纹识别、近场通信 (NFC)、红外线操控这些需要特殊传感器设计的，那对测试设备和平台的选择就相对少一些，只需要考虑那些拥有这些传感器的设备。例如对于支持指纹识别的 App，测试人员需要考虑的设备也就是 iPhone 5s、iPhone 6、iPhone 6Plus、iPad Air2、iPad mini3、LG G3、三星 Galaxy S5、三星 Galaxy Note4、HTC One Max 和华为 Mate7 这些设备（不考虑市场占有率比较低的 vivo 和 OPPO 的手机）；而如果 App 支持心率监测，测试人员就只能选择三星 Galaxy S5 和 Galaxy Note4 了。

这里推荐大家使用一个网站 (<http://www.phonearena.com>) 来做设备的查找工作。通过这个网站不仅可以查询到各种手机和平板设备的详细参数信息，还可以对它们进行横向对比，方便测试人员找到适合用来做测试的设备（如图 1.1 所示）。

(2) 如果 App 是针对某种平台所独有的功能设计的，或者是某种平台独占的，测试人员就只需要考虑相应平台下的设备。比如 App 是类似 Android 设备上层出不穷的“xx 清理大师”，那在确定测试设备和平台时就不需要考虑 iOS 平台了；又比如之前 Instagram 选择只支持 iOS 平台，那作为 Instagram 的测试人员只需要关注于 iOS 设备就足够了。


RIVALS			
	Sony Xperia Z2		
	Samsung Galaxy S4		
	Samsung Galaxy Note 3		
	HTC One (M8)		
		Samsung Galaxy S5	Apple iPhone 6
		Posted: 24 Feb 2014 Market Status: Released US carrier availability: Available - Verizon, AT&T, Sprint, T-Mobile, MetroPCS, U.S. Cellular, Cricket, Virgin Mobile, Boost Mobile	Posted: 09 Sep 2014 Market Status: Released US carrier availability: Available - Verizon, AT&T, Sprint, T-Mobile
		PhoneArena rating: 9.1	PhoneArena rating: 9
		User rating: 9.2	User rating: 9.4
			Google Nexus 5
			Posted: 31 Oct 2013 Market Status: Released US carrier availability: Available - Sprint, T-Mobile, Upcoming - AT&T
			PhoneArena rating: 9.5
			User rating: 9.4
DESIGN			
Device type	Smart phone	Smart phone	Smart phone
OS	Android (4.4.2) TouchWiz UI	iOS (8.x)	Android (4.4.4, 4.4.2, 4.4)
Dimensions	 5.59 x 2.85 x 0.32 inches (142 x 72.5 x 8.1 mm)	 5.44 x 2.64 x 0.27 inches (138.1 x 67 x 6.9 mm)	 5.43 x 2.72 x 0.34 inches (137.84 x 69.17 x 8.59 mm)
Weight	 5.11 oz (145 g) the average is 4.9 oz (139 g)	 4.55 oz (129 g) the average is 4.9 oz (139 g)	 4.59 oz (130 g) the average is 4.9 oz (139 g)
Body material		Aluminium	
Rugged	Water, Dust resistant		
IP certified	IP 67		
DISPLAY			
Physical size	 5.1 inches	 4.7 inches	 5.0 inches
Resolution	 1080 x 1920 pixels	 750 x 1334 pixels	 1080 x 1920 pixels
Pixel density	 432 ppi	 326 ppi	 445 ppi
Technology	Super AMOLED	IPS LCD	IPS LCD
Colors	16 777 216	16 777 216	16 777 216
Touchscreen	Multi-touch	Multi-touch	Multi-touch
Features	Light sensor, Proximity sensor, Scratch-resistant glass	Light sensor, Proximity sensor, Scratch-resistant glass, Oleophobic coating	Light sensor, Proximity sensor, Scratch-resistant glass

图 1.1 <http://www.phonearena.com> 设备对比

或者，如果 App 选择不支持某种平台，相应的，测试人员也就不需要测试运行这些平台的设备了。比如 WindowsPhone、黑莓（BlackBerry）以及塞班（Symbian）平台在市场上的占有率已经很低了（根据 2014 年第四季度的调查，详见图 1.2），如果

在开发时选择不支持这些平台，那在测试时测试人员就完全可以忽略相关的设备。

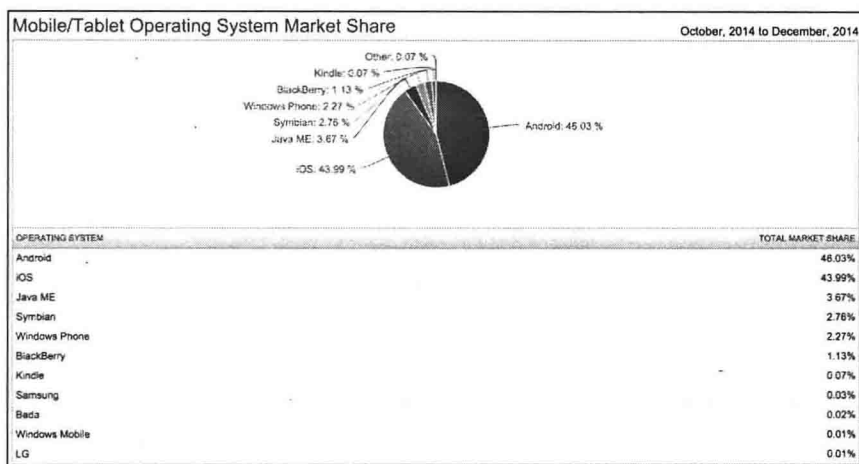


图 1.2 2014 年第四季度各操作系统市场占有率调查表（数据来源：www.netmarketshare.com）

(3) 如果 App 是面向大众的通用型 App，测试人员就需要结合移动 App 的生命周期和测试设备的硬件参数来确定测试设备和平台了。

1.2 移动 App 的生命周期

(1) 对于还处于开发阶段但准备不久之后投入市场的一款新 App，鉴于并没有已经实际使用 App 的用户，所以测试人员要“预测”真实的用户所使用的设备和平台。在这种情况下，首先需要了解使用 App 的主要用户是哪一类人群，比如说是发烧友，还是商务人士。发烧友极有可能使用的是最新的设备和平台；商务人士更多使用的是成熟的平台，高端一些的设备；而如果用户是普通大众，就需要通过 Apple 和 Google 官方发布的版本占有率数据来帮助测试人员进行有依据的“拍脑袋”了。

以下是 Apple 官方发布的 iOS 版本占有率数据 (<https://developer.apple.com/support/appstore/>)，如图 1.3 所示；和 Google 官方发布的 Android 版本占有率数据 (<http://developer.android.com/about/dashboards/index.html>)，如图 1.4 所示。

(2) 对于已经发布并且有稳定用户群的 App，测试人员可以使用在桌面应用开发时用到的工具，例如 Google Analytics 或 Omniture SiteCatalyst（现在 Omniture 被 Adobe 收购了，工具也改名叫做 Adobe Analytics）来统计用户的信息，从而确定 App 支持和

需要测试的设备及平台。这里对于 App 有一点要求，就是 App 需要联网对后台的服务器发送请求，从而能获取到用户信息。

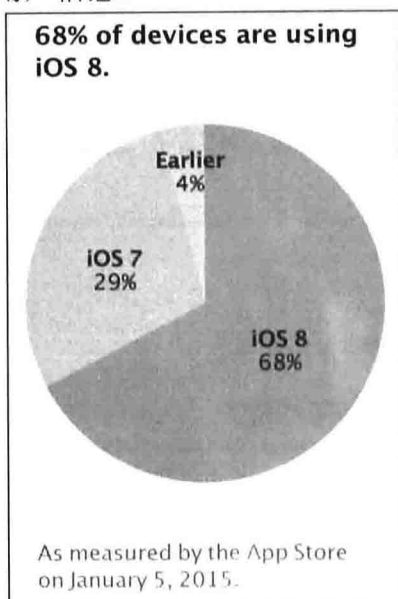


图 1.3 截至 2015 年 1 月 5 日，iOS 各版本所占比例

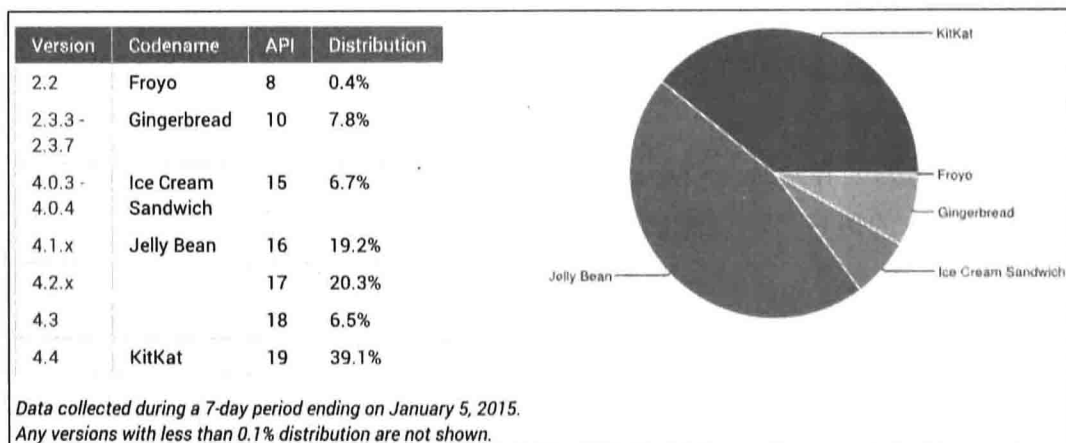


图 1.4 截至 2015 年 1 月 5 日，Android 各版本所占比例

Google Analytics (Google 分析, 网址为 <http://www.google.com/analytics>) 是 Google 的一款免费的网站分析服务, 使用范围十分广泛。Google Analytics 功能非常强大, 只要在网站的页面上加入一段代码, 就可以提供丰富详尽的图表式报告。Google Analytics 的特点是简单易用, 但是相应的缺点就是不可定制化。Google Analytics 的页面如图 1.5 所示。

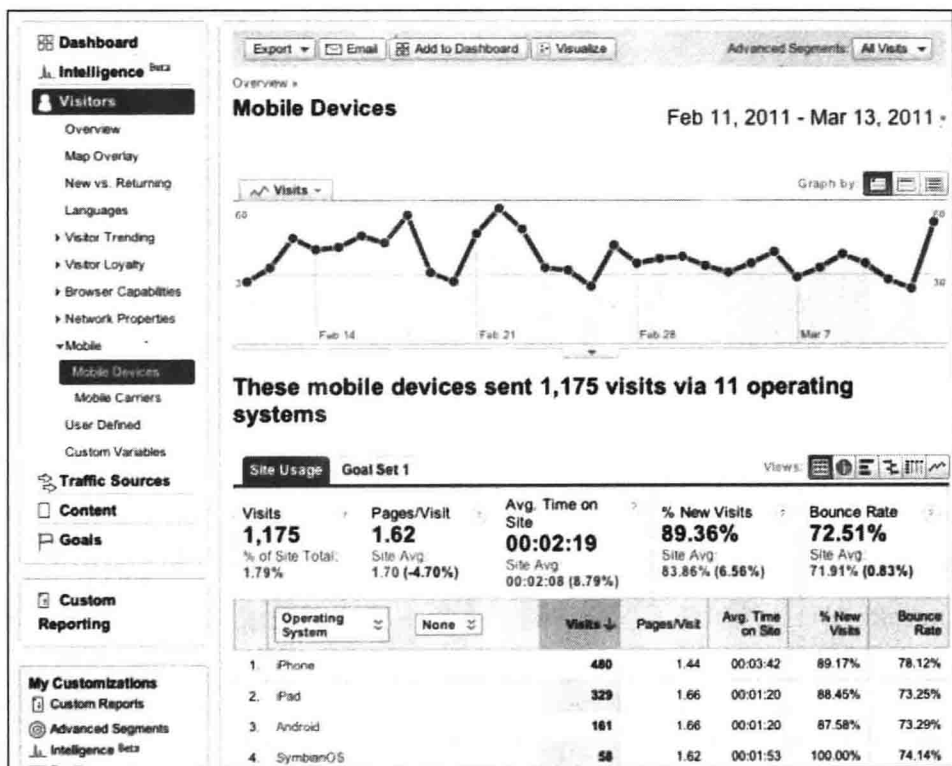


图 1.5 Google Analytics

Omniure SiteCatalyst (Adobe Analytics, 网址为 <http://www.adobe.com/solutions/digital-analytics.html>) 是一个进行网站基本指标的搜集、报告和分析的工具。通过这个软件可以得到网站和 App 的访问量、浏览量、跳出率、转化率、来源等诸多指标。只要在 App 中对不同事件以及发送请求都添加相应的 Omniure 追踪, 然后再登录 Omniure 的网页就可以进行用户数据分析。Omniure SiteCatalyst 不同于 Google Analytics 的一个特点是, 它可以对数据进行高级细分, 也就是说, 可以对用户的各种操作打上不同的标签, 在服务器端搜集到信息后进行统一的筛选和分析。Omniure SiteCatalyst 的页面如图 1.6 所示。

(3) 对于上面两种情况, 有一种特例需要考虑, 就是在有新的操作系统版本将要发布的时候, 需要参考以前操作系统版本升级时用户更新的进度。正如图 1.3 和图 1.4 所示, 在 iOS 8 发布 3 个月之内有 68% 的用户进行了升级, 而使用 iOS 7 之前版本的用户只有 4%; 而 Android 4.4 Kitkat 发布一年后, 市场占有率才刚刚达到 39.1%, 有超过 52.7% 的用户使用的还是 4.0~4.3 版本的 Android, 甚至还有 8.2% 左右的用户还在使用着 Android 2.x 的设备。