测试人员常被看作Bug寻找者，但你曾想过他们实际是如何开展测试的吗？你是否好奇他们究竟都做些什么，以及他们如何在一个典型的技术项目中体现价值？

作者将带你经历测试人员的思维过程，探讨他们测试移动App时的各种考虑。本文的目的在于揭示测试人员的这一思维过程，并展示他们通常所考虑内容的广度和深度。

#### **测试人员需要询问问题**

测试人员的核心能力在于提出有挑战性的相关问题。如果你能将调查、询问技巧和技术、产品的知识结合起来，渐渐地，你也会成为一个好的测试人员。

比如，测试人员可能会问：

* 这个App应该在什么平台上使用？
* 这个App到底是干什么的？
* 如果我这样做，会发生什么情况？

诸如此类。

测试人员能从各种场景中发现问题，它们可能来自对话、设计、文档、用户反馈或者是产品本身。这些可能性太多了……因此，让我们一探究竟吧！

#### **从哪里开始测试**

理想情况下，测试人员应该掌握所测产品的所有最新细节资料。但事实上这很少见，因此，像其他人一样，测试人员只能将就使用手上有限的资料。但这不是不能测试的借口！测试人员其实是可以从内部和外部多种不同的来源处收集信息的。

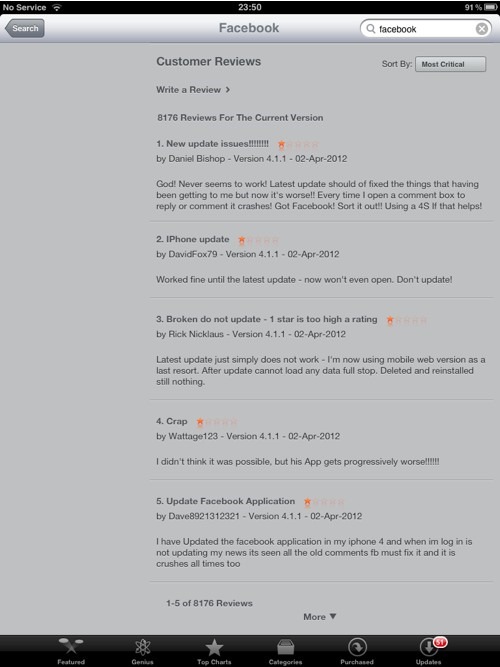
这个阶段，测试人员可以问这些问题：

* 有哪些信息：规格？项目会议？用户文档？知识渊博的团队成员？有支持论坛或者是公司在线论坛提供帮助？有现存Bug的记录吗？
* 该应用是在什么系统、平台和设备上进行运作和测试？
* 该应用是处理什么类型的数据（比如个人信息、信用卡等等）？
* 该应用有整合外部应用（比如API和数据来源）吗？
* 该应用需要用到特定的移动端网页吗？
* 现有消费者如何评价这个产品？
* 有多少时间可用于测试？
* 测试的优先级和风险是什么？
* 哪些用户使用起来不愉快，为什么？
* 如何发布和更新？

基于以上收集的信息，测试人员可以制定测试计划了。通常预算决定测试方法，一天测完，一个星期或一个月测完的方法肯定不同。当你逐渐熟悉团队、工作流程以及这类问题的解决方式时，你就更容易预测结果了。

案例：FacebookApp的社会评论

当作为一名测试人员收集信息时，我喜欢选用FacebookApp作为案例，因为用户的抱怨到处都是。以下仅仅展示了部分遇到难题的用户在iTunesAppStore中发表的评论，网络上还有很多。



iPhone上的FacebookApp有很多负面的评论

如果我接受挑战去测试Facebook这个App，我肯定会考虑这些反馈，否则就是傻子。

#### **测试人员的创造力**

你可能知道这个App原本想做的事，但是它究竟可以做什么事呢？用户实际上是如何使用它的？测试人员擅长作为旁观者来思考，尝试不同的事物，以及不断地询问“如果。。。会怎么样”和“为什么”的问题。

比如，移动端的测试人员常常以不同的用户角色进行测试——当然有点夸张，但是，这种把自己当成不同用户进行思考、分析和设想的能力对测试是备受启发的。

测试人员可能会设想自己是以下用户：

* 毫无经验；
* 很有经验；
* 爱好者；
* 黑客；
* 竞争对手；

当然还有更多可选的角色，这主要取决于你们所开发的产品是什么。其实除了角色特点外，其操作行为和工作流程也很重要。人们使用产品方式常常很奇怪，比如：

* 在不应该返回的时候返回了；
* 不耐心而且多次敲按键；
* 输入错误的数据；
* 不理解该怎么做；
* 可能没有按要求进行设置；
* 可能会自以为是地认为自己知道该做什么（比如通常不阅读说明）。

测试人员遇到这些问题时，也常常发现意料之外的Bug。有时候，这些Bug微不足道，但是**更深入的调查就会发现更严重的问题**。

很多问题是可以被预先确定和测试的。测试移动端App时，以下的问题并不都有关，但是也可以尝试问问：

* 是否按照所说的来做呢？
* 是按设计完成任务的吗？
* 不是按设计完成任务的吗？
* 如果处于一直被使用或者负荷情况下，状况会怎么样？会反应迟钝吗？会崩溃吗？会更新吗？有反馈吗？
* 崩溃报告会反馈到App吗？
* 用户可能有哪些创造性的、逻辑性的或是消极的导航方式？用户相信你的品牌吗？
* 用户的数据安全如何？
* 有可能被中断或是被破解吗？
* 运行到极限时会发生什么状况？
* 会要求打开相关服务吗(如GPS、Wi-Fi)？如果用户打开会怎样？没打开又会怎样？
* 将用户重新引向哪儿？去网页？还是从网页到App？这会导致问题出现吗？
* 沟通过程和市场反馈是否符合该App的功能、设计和内容？
* 登录流程是怎样的？能在App上直接登录还是要去网页端？
* 登录是否整合了其他服务，比如用Facebook和Twitter帐号登录？

案例：RunKeeper’sgyUpdate

RunKeeper，是一款能跟踪你健身活动的App，最新发布的版本里有个“目标设置”的功能，对此我很感兴趣去体验一下，一部分从测试人员的角度来看，更多的是作为一个真心喜欢产品的用户来体验。但我发现了一些问题：

1.默认单位是英镑，我却想要把公斤作为重量单位；

2.英镑和公斤间的切换根本不好用；

3.当设定目标后，会导致展示错误的数据和图表，这让我很迷惑；

4.由于第3条，我想删除目标，但却根本找不到删除的地方；

5.为了解决这一问题，我不得不改变的个人体重的值，直到“目标设置“范围之内，这样目标达到了，就能重新设定目标了；

6.我会再次尝试添加目标；

正因为以上疑惑，我花了更长的时间把玩它，看能不能找到其他的问题；

以下是一些发现问题的屏幕截图：



该App的最新版本包含了一个新的“目标”部分。设置日期的时候，我发现开始和结束的日期都可以从公元1年开始，另外，为什么有两个1年可选（译者注：年份那列从上往下应该显示为“1、2、3”）？



另一个Bug，是“当前体重”部分的一个拼写错误，当清空数据时会出现拼写错误的“Enter“（应用中用的是Etner），这只是一个小Bug，但是看上去非常不专业。

发现问题没有捷径，你只能反复的慢慢的试用。每个App及其团队都会面临很多不同的挑战。但是，测试人员的典型的特点就是：超越极限，做一些非常规的、可以改变周围事物的事情，保持长时间的测试（测试几天、几个星期甚至几月，而不是几分钟就测完），即使明明知道这些事情是不可能发生的。这些也正是可以找到和引出的场景所在。

#### **哪儿有所有的数据？**

测试人员喜欢从数据上找问题，这让开发人员有时候很郁闷。事实上，用户或者是软件开发人员在信息流中确实太容易迷惑了，因为可能会出现很多错误，所以基于数据和云的服务更为重要。

也许你可以尝试在以下场景中检查出问题：

* 移动设备数据已满；
* 测试人员移除了所有的数据；
* 测试人员删除了App，那数据怎么办？
* 测试人员删除并重装了App，数据怎么办？
* 过多或者过少的内容导致设计和布局的改变；
* 在不同的时间段和时区使用；
* 数据不同步；
* 同步被中断；
* 数据更新影响其他的服务（比如网页和云端服务）；
* 快速处理数据或是处理大量的数据；
* 使用无效的数据；

案例：Soup.me的错误

我试用过的Soup.me，是一个可以通过地图和颜色将个人Instagram中的照片进行分类的网页服务，但是我却没用多久。当注册时，它提示我Instagram上的照片不够多，然而我的账号中明明有500多张照片。我并不清楚问题出在哪儿，也许是数据问题，也许是表现层的问题，也有可能是该App出错提示的问题。



另一个案例：Quicklytics

Quickytics是一个iPad上的网页分析应用。在使用过程中，尽管我已经从Google Analytics中删除了网站配置，但它仍然存在。这里有一些问题：

我已经删除了网站配置，为什么还是有这些信息？

左边模块没有解释为什么“该操作无法完成”，那么是不是可以改进以避免迷惑用户呢？



测试人员也很喜欢测试极限数据下的情况。他们常常是作为典型用户来了解这个App，所以极限下的测试并不会花很长的时间。数据是混乱的，所以测试人员要考虑到软件的用户类型，以及在不同的数据场景下如何进行测试。

比如，他们可能尝试以下场景：

* 测试用户可输入的极限值；
* 用重复的数据进行测试；
* 在全新无数据的手机里测试；
* 在老手机上测试；
* 预先安装不同类型的数据；
* 考虑聚集大家的资源来进行测试；
* 让一些测试自动化；
* 用一些超出预期的数据去测试，看它是怎么处理的；
* 分析信息和数据是怎么影响用户体验的；
* 不管用户看到的是否正确，都要一直问问题。

#### **创建出错提醒和消息**

这里，我不是从设计师的角度来要谈论好的错误消息的设计，而是想从用户或是测试者的角度来看这个问题。出错提醒和消息是测试人员很容易发现问题的地方。

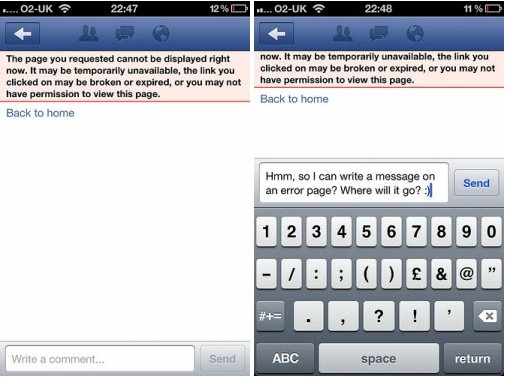
关于错误信息要问的问题：

* 请考虑以下问题：
* 出错提醒的UI设计可以接受吗？
* 错误信息内容可以理解吗？
* 错误信息是否保持一致？
* 这些错误信息有帮助吗？
* 错误信息内容是否合适？
* 这些错误是否符合惯例和标准？
* 这些错误信息本身是否安全？
* 运行记录和崩溃是否能被用户和开发者获得？
* 是否所有的错误都被测试过？
* 用户处理完错误信息后，将处于什么状态
* 是否在用户应该接受错误信息时，却没有错误信息弹出？

错误信息会影响用户体验。然而，**不好或无用的出错提醒无处不在。**虽最理想的状态是避免用户遭遇错误信息，但这几乎不可能。出错情况的设计、实现和确认可能与预期相反，但是，测试者往往善于发现意料外的Bug，并能仔细考究是否改进它们。

错误信息的案例：

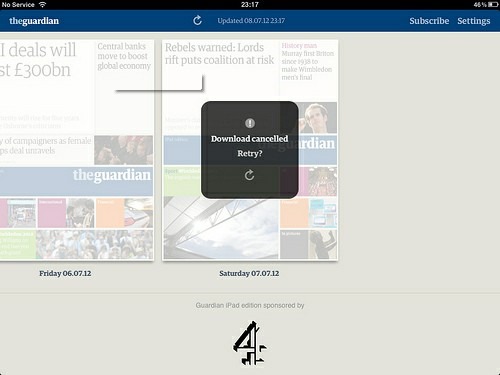
我非常喜欢举iPhone上FacebookApp这个例子。这些冗长又晦涩的文字不仅仅试图涵盖了许多不同的场景，而且还可能无端地丢失。



可能如下的消息提示框可以列入反例“名人堂”了？



看看这款iPad上的The Guardian应用，如果我不想“重试”，该怎么办呢？



#### **特定平台上的注意事项**

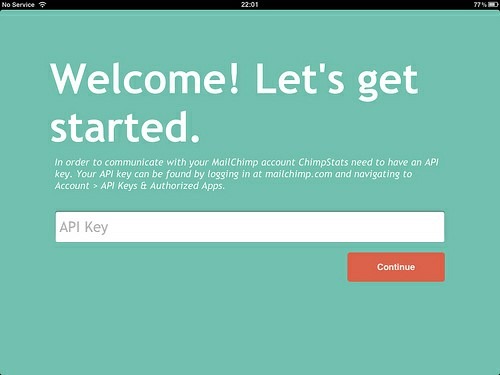
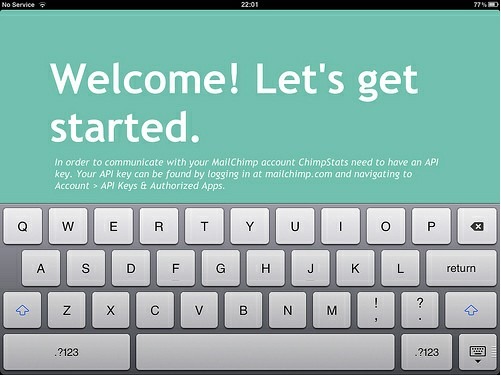
对于任何项目团队成员来说，了解相关平台的业务、技术和设计上的限制，都是至关重要的。

那么，移动端App的测试人员应该找出哪些平台相关的问题呢？

* 是否遵照了这个特定平台的设计规范？
* 与竞争对手以及行业内的设计相比如何？
* 是否适应外围设备？
* 触摸屏支持手势吗，如：轻拍、双击、长按、拖动、摇动、夹捏、轻拂、滑动？
* 这个App可以被理解吗？
* 当转动设备的方向时，有什么变化？
* 可以使用地图和GPS吗？
* 有用户指南吗？
* 电子邮件的工作流程友好吗？
* 通过网络分享时，它运行得流畅吗？是否整合了其他社交应用或网站？
* 当用户正在进行多任务工作，并在不同App间切换的时候，它还运行正常吗？
* 当用户更新它时，它是否会显示时间进度？
* 默认设置如何?有经过调整吗？
* 使用音效会有不同吗？

案例：ChimpStats

ChimpStats是iPad上一个查看邮件广告详情的应用。我第一次使用这个应用是处于横屏模式。当我需要输入API密码的时候，我被困住了。我根本不能在水平模式中输入API密码，直到切换成竖屏模式，才输入成功。

连接和中断的问题当连接断断续续或是意外中断时，很多有趣的事情就可能发生了。

你是否尝试过在以下场景中使用App：

* 走动环境下？
* Wi-Fi连接下？
* 没有Wi-Fi的情况下？
* 3G模式下？
* 间歇性地连接？
* 设置为飞行模式？
* 一个电话打进来时？
* 接收到一条信息时？
* 接收到一个提醒通知时？
* 在电量很低甚至自动关机时？
* 被强制更新时？
* 收到一条语音留言时？

这类测试最容易发现错误和Bug。我强烈建议你在这些情况下进行测试（不仅仅只是开机、确认它可以正常工作，还要尝试用户使用的整个流程，并在特定的时间间歇内强制连接和中断）。

* 这个App提供了足够多的反馈吗？
* 数据传输为用户所知吗？
* 它会慢慢停止，然后崩溃吗？
* 开启时会发生什么？
* 任务完成中会发生什么？
* 是否可能丢失未保存的操作？
* 你可以忽视通知提醒吗？忽视后会发生什么？
* 你可以对通知提醒做出响应吗？响应后会发生什么？
* 对某些问题，使用错误信息是否恰当？
* 当登录过期或超时会发生什么？

#### ****App的维护****

想要加快整个测试的过程很简单，只需测试一次就一劳永逸了，对吗？请三思。

此刻我遇到的一个问题是：iPad上的一些App在更新后，再也不能下载了。对于一个用户来说，这是非常令人沮丧的。

可能，这也是开发者控制不了的。谁知道呢？我只知道它对于用户来讲是不能用的。我也尝试卸载App，然后重装，但这个问题始终未能解决。我在网上大量的搜索，除了找到一些关于更新操作系统的建议外，没有任何其他解决方式。可能，下次有空时候，我还会再试试看。

关键问题在于：如果一个应用只被测试过一次，且只有一次（或仅在很短的一段时间内测试过），很多问题你都发现不了。一个App自身可能不会发现变化，但外界条件却可以让这些问题发生。

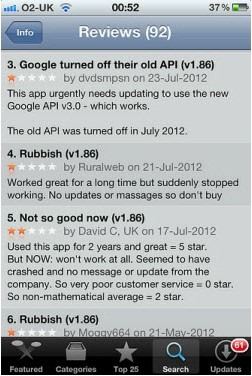
当外界环境持续变化时，App又会受到哪些影响呢？让我们问问自己：

* 我可以下载这个App吗？
* 我可以下载并安装更新吗？
* 更新之后还能使用吗？
* 当很多App处于等待更新状态时，我能更新它吗？
* 系统更新后，它会发生什么？
* 系统未更新，它又会发生什么？
* 它会通过iTunes自动同步下载到其他设备吗？
* 它自动执行任务或测试有意义吗？
* 它会连接到网络服务吗？这会带来什么不同？

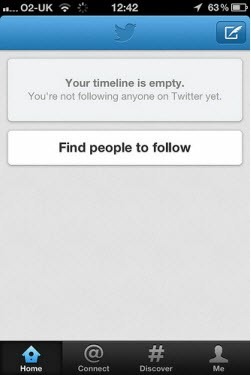
移动端的App每一个版本发布后，最好都去测试一下。每次发布新版本时，先定义最高优先级测试，确保其能在各种条件下进行（主要是在主流的平台上）。随着时间的推移，测试可以变得自动化。但请记住，自动化不是灵丹妙药，发现问题，只能通过人的眼睛。

案例：iPhone上的Analytics应用

我使用这个App已经两年了，之前它一直没有什么问题。但是现在，它却显示出我某些网站数据为零（但实际上，不止一个人一个月内访问过我的网站！）。从App Store的评论来看，我不是唯一一个遇到这个问题的人。

另外一个案例是iPhone上的Twitter。更新并启动这个App后，我瞬间看到了如下这个提示语：“你的时间线数据显示为空，你至今没有关注任何人”(但我是拥有5年经验的活跃用户)。我担心了一会儿，庆幸的是，这个消息很快就消失，然后加载出历史数据。



#### **测试不是对错判断**

我们讨论了移动测试的一些方面，但这些前提是：**带着问题，才能发现问题。**

通常，测试被认为是完全合乎逻辑的、可计划的和可预测的，过程包括：测试脚本和测试计划、通过和失败、正确和错误的反馈。走完这些测试流程就离真相不远了。

当然，如果必要，我们可以用上述方法进行测试，但这并不是测试的目的。我们不仅是为了创建测试用例、发现Bug，更重要的是找到关键的问题，为项目组决定什么时候发布App提供有价值的信息。而找到那些关键问题的最好方法就是：提问！