

APP 测试流程 & 测试资料合集 V3.0

注：2019/07/19 更新 V3.0 版本

适用对象：

刚开始接触 APP 测试，或者对 APP 测试不是特别了解的初中级测试同学。

比如，

- 1) 虽然干过多年测试，但是从来没玩过 APP 测试。突然，某一天，公司让你测试 APP。那么，这份资料，足够入行了，看完可以直接干活。
- 2) 刚刚入行测试，对 APP 完全不了解，那就更适合了（也许，你还可以去看看老徐之前的 Web 测试，看完，基本上常见测试类型，就掌握了）

此份资料，版权归 IDO 老徐所有，仅限个人阅读 & 参考，禁止打印 & 商用。

作者：IDO 老徐 来源：公众号「简尚」

博客 <http://isTester.com>

正文：

/ 1 /

测试理论知识：跟其他 web 等测试一样（Web 测试资料，可以在老徐的资料合集里面找到 <http://istester.com/tester/243.html>）；

如果基础不够老徐建议先看书: [给软件测试从业者推荐 6 本书籍](#)

/ 2 /

整体测试流程: 同 web 类似

具体差异化见下方内容

/ 3 /

App 独特测试点:

客户端兼容性测试: 系统版本、不同深度定制的 rom、屏幕分辨率、中断测试、安装、卸载、升级、对其他程序的干扰等

需要的一些工具:

appium / lR / jmeter / 打包工具 / 抓包工具等

/ 4 /

android & ios 区别?

没啥区别, 基本就是一些系统特性不一样, 多数相同

特性测试 (横竖屏, home 键, 音量键, 返回, power 键等)

/ 5 /

在测试的过程中, 学会利用各种测试辅助工具可以减少我们的重复性动作, 提高效率, 让测试事半功倍。

有很多云测工具: 如 testin/ mqc / WeTest 等

如果你在测 App, 应该知道是什么, 有什么用。如果不知道的, 搜索引擎, 检索下, 便知。

/ 6 /

是否需要搭建环境？

基本不需要，服务端统一搭建部署服务，跟在某个客户端运行关系不大，无需单独部署环境

/ 7 /

App 打包，上传到应有市场

如果你测 App 应该知道这些：

给大家推荐两个工具网站

fir.im/

www.pgyer.com/

OK，下面是具体的细化测试点：

老徐花了几个小时收集、整理、部分筛选自网络

IDO 老徐，注：

这些测试点，可以直接拿来用，根据你们 APP 的特别，补充一些即可。

/ 1 /

软件权限

1) 扣费风险：包括发送短信、拨打电话、连接网络等

- 2) 隐私泄露风险：包括访问手机信息、访问联系人信息等
- 3) 对 App 的输入有效性校验、认证、授权、敏感数据存储、数据加密等方面进行检测
- 4) 限制/允许使用手机功能接入互联网
- 5) 限制/允许使用手机发送接受信息功能
- 6) 限制/允许应用程序来注册自动启动应用程序
- 7) 限制或使用本地连接
- 8) 限制/允许使用手机拍照或录音
- 9) 限制/允许使用手机读取用户数据
- 10) 限制/允许使用手机写入用户数据
- 11) 检测 App 的用户授权级别、数据泄漏、非法授权访问等

/ 2 /

安装与卸载安全性

- 1) 应用程序应能正确安装到设备驱动程序上
- 2) 能够在安装设备驱动程序上找到应用程序的相应图标
- 3) 是否包含数字签名信息
- 4) JAD 文件和 JAR 包中包含的所有托管属性及其值必需是正确的
- 5) JAD 文件显示的资料内容与应用程序显示的资料内容应一致
- 6) 安装路径应能指定
- 7) 没有用户的允许，应用程序不能预先设定自动启动
- 8) 卸载是否安全，其安装进去的文件是否全部卸载
- 9) 卸载用户使用过程中产生的文件是否有提示
- 10) 其修改的配置信息是否复原

11) 卸载是否影响其他软件的功能

12) 卸载应该移除所有的文件

/ 3 /

数据安全性

1) 当将密码或其他的敏感数据输入到应用程序时，其不会被储存在设备中，同时密码也不会被解码

2) 输入密码将不以明文形式进行显示

3) 密码，信用卡明细，或其他的敏感数据将不被储存在它们预输入的位置上

4) 不同的应用程序的个人身份证或密码长度必需至少在 4—8 个数字长度之间

5) 当应用程序处理信用卡明细，或其他的敏感数据时，不以明文形式将数据写到其它单独的文件或者临时文件中。以防止应用程序异常终止而又没有删除它的临时文件，文件可能遭受入侵者的袭击，然后读取这些数据信息。

6) 当将敏感数据输入到应用程序时，其不会被储存在设备中

7) 备份应该加密，恢复数据应考虑恢复过程的异常通讯中断等，数据恢复后再使用前应该经过校验

8) 应用程序应考虑系统或者虚拟机器产生的用户提示信息或安全警告

9) 应用程序不能忽略系统或者虚拟机器产生的用户提示信息或安全警告，更不能在安全警告

/ 4 /

图形测试

1) 横向比较。整个 App 各控件操作方式统一

2) 自适应界面设计，内容根据窗口大小自适应

3) 页面标签风格是否统一

- 4) 页面是否美观
- 5) 页面的图片应有其实际意义而要求整体有序美观
- 6) 图片质量要高且图片尺寸在设计符合要求的情况下应尽量小
- 7) 界面整体使用的颜色不宜过多

老徐备注：其实跟 web 类似，通用的测试点；

/ 5 /

功能测试

此处只提炼一些差异性的内容；基本同 web 等其他类型测试

具体见 Web 测试资料

/ 6 /

运行

- 1) App 安装完成后的试运行，可正常打开软件。
- 2) App 打开测试，是否有加载状态进度提示。
- 3) App 打开速度测试，速度是否可观。
- 4) App 页面间的切换是否流畅，逻辑是否正确
- 5) 注册

同表单编辑页面

用户名密码长度

注册后的提示页面

前台注册页面和后台的管理页面数据是否一致

注册后，在后台管理中页面提示

- 6) 登录

使用合法的用户登录系统。

系统是否允许多次非法的登陆，是否有次数限制。

使用已经登陆的账号登陆系统是否正确处理。

使用禁用的账号登陆系统是否正确处理。

用户名、口令（密码）错误或漏填时能否登陆。

删除或修改后的用户，原用户登陆。

不输入用户口令和用户、重复点（确定或取消按钮）是否允许登陆。 一登陆后，页面中登陆信息。

页面中有注销按钮。

登陆超时的处理。

老徐备注：如上基本上是登陆的通用测试点，学会提炼

7) 注销

注销原模块，新的模块系统能否正确处理。

终止注销能否返回原模块，原用户。

注销原用户，新用户系统能否正确处理。

使用错误的账号、口令、无权限的被禁用的账号进行注销

/ 7 /

应用的前后台切换

- 1) APP 切换到后台，再回到 app，检查是否停留在上一次操作界面。
- 2) APP 切换到后台，再回到 app，检查功能及应用状态是否正常
- 3) app 切换到后台，再回到前台时，注意程序是否崩溃，功能状态是否正常，尤其是对于从后台切换回前台数据有自动更新的时候。
- 4) 手机锁屏解屏后进入 app 注意是否会崩溃，功能状态是否正常，尤其是对于从后台切换回前台数据有自动更新的时候。

- 5) 当 App 使用过程中有电话进来中断后再切换到 app, 功能状态是否正常
- 6) 当杀掉 app 进程后, 再开启 app, app 能否正常启动。
- 7) 出现必须处理的提示框后, 切换到后台, 再切换回来, 检查提示框是否还存在, 有时候会出现应用自动跳过提示框的缺陷。
- 8) 对于有数据交换的页面, 每个页面都必需要进行前后台切换、锁屏的测试, 这种页面最容易出现崩溃。

/ 8 /

免登录

很多应用提供免登录功能, 当应用开启时自动以上一次登录的用户身份来使用 app.

- 1) app 有免登录功能时, 需要考虑 IOS 版本差异。
- 2) 考虑无网络情况时能否正常进入免登录状态。
- 3) 切换用户登录后, 要校验用户登录信息及数据内容是否相应更新, 确保原用户退出。
- 4) 如果一个帐户只允许登录一台机器。需要检查一个帐户登录多台手机的情况。原手机里的用户需要被踢出, 给出友好提示。
- 5) app 切换到后台, 再切回前台的校验
- 6) 切换到后台, 再切换回前台的测试
- 7) 密码更换后, 检查有数据交换时是否进行了有效身份的校验
- 8) 支持自动登录的应用在进行数据交换时, 检查系统是否能自动登录成功并且数据操作无误。
- 9) 检查用户主动退出登录后, 下次启动 app, 应停留在登录界面

/ 9 /

数据更新

根据应用的业务规则,以及数据更新量的情况,来确定最优的数据更新方案。

- 1) 需要确定哪些地方需要提供手动刷新,哪些地方需要自动刷新,哪些地方需要手动+自动刷新。
- 2) 确定哪些地方从后台切换回前台时需要进行数据更新。
- 3) 根据业务、速度及流量的合理分配,确定哪些内容需要实时更新,哪些需要定时更新。
- 4) 确定数据展示部分的处理逻辑,是每次从服务端请求,还是有缓存到本地,这样才能有针对性的进行相应测试。
- 5) 检查有数据交换的地方,均有相应的异常处理。

/ 10 /

离线浏览

很多应用会支持离线浏览,即在本地客户端会缓存一部分数据供用户查看。

- 1) 在无网络情况可以浏览本地数据
- 2) 退出 app 再开启 app 时能正常浏览
- 3) 切换到后台再切回前台可以正常浏览
- 4) 锁屏后再解屏回到应用前台可以正常浏览
- 5) 在对服务端的数据有更新时会给予离线的相应提示

/ 11 /

App 更新

- 1) 当客户端有新版本时,有更新提示
- 2) 当版本为非强制升级版时,用户可以取消更新,老版本能正常使用。用户在下次启动 app 时,仍能出现更新提示。
- 3) 当版本为强制升级版时,当给出强制更新后用户没有做更新时,退出客户端。下次启动 app 时,仍出现强制升级提示。

- 4) 当客户端有新版本时，在本地不删除客户端的情况下，直接更新检查是否能正常更新。
- 5) 当客户端有新版本时，在本地不删除客户端的情况下，检查更新后的客户端功能是否是新版本。
- 6) 当客户端有新版本时，在本地不删除客户端的情况下，检查资源同名文件如图片是否能正常更新成最新版本。

如果以上无法更新成功的，也都属于缺陷。

- 7) 测试 App 的增量升级，测试 App 在升级失败以后，是否可以继续使用升级之前的版本。
- 8) 新版发布后，配合不同网络环境的自动更新提示及下载、安装、更新、启动、运行的验证测试。

需要检查的内容：

测试升级后的功能是否与需求说明一样

测试与升级模块相关的模块的功能是否与需求一致

升级安装意外情况的测试（如死机、断电、重启）

升级界面的 UI 测试

不同操作系统间的升级测试

数字签名、升级覆盖安装、下载后手动覆盖安装、跨版本升级、升级后可以正常使用。覆盖安装要确保数据库有字段更新的话，能正常更新，否则就容易导致 app 异常。

/ 12 /

定位、照相机服务

- 1) App 有用到相机，定位服务时，需要注意系统版本差异
- 2) 有用到定位服务、照相机服务的地方，需要进行前后台的切换测试，检查应用是否正常。

- 3) 当定位服务没有开启时, 使用定位服务, 会友好性弹出是否允许设置定位提示。当确定允许开启定位时, 能自动跳转到定位设置中开启定位服务。
- 4) 测试定位、照相机服务时, 需要采用真机进行测试。

/ 13 /

时间测试

客户端可以自行设置手机的时区、时间, 因此需要校验该设置对 app 的影响。

中国为东 8 区, 所以当手机设置的时间非东 8 区时, 查看需要显示时间的地方, 时间是否展示正确, 应用功能是否正常。时间一般需要根据服务器时间再转换成客户端对应的时区来展示, 这样的用户体验比较好。

比如发表一篇微博在服务端记录的是 10: 00, 此时, 华盛顿时间为 22: 00, 客户端去浏览时, 如果设置的是华盛顿时间, 则显示的发表时间即为 22:00, 当时间设回东 8 区时间时, 再查看则显示为 10: 00。

/ 14 /

PUSH 测试

- 1) 检查 push 消息是否按照指定的业务规则发送
- 2) 检查不接受推送消息时, 检查用户不会再接收到 push.
- 3) 如果用户设置了免打扰的时间段, 检查在免打扰时间段内, 用户接收不到 PUSH。在非免打扰时间段, 用户能正常收到 push。
- 4) 当 push 消息是针对登录用户的时候, 需要检查收到的 push 与用户身份是否相符, 没有错误地将其它人的消息推送过来。

一般情况下, 只对手机上最后一个登录用户进行消息推送。

- 5) 测试 push 时, 需要采用真机进行测试。

/ 15 /

性能测试

评估 App 的时间和空间特性：

1) 极限测试：在各种边界压力情况下，如电池、存储、网速等，验证 App 是否能正确响应。

内存满时安装 App --运行 App 时手机断电 --运行 App 时断掉网络

2) 响应能力测试：测试 App 中的各类操作是否满足用户响应时间要求。 --App 安装、卸载的响应时间 --App 各类功能性操作的影响时间

3) 压力测试：反复/长期操作下、系统资源是否占用异常。

App 反复进行安装卸载，查看系统资源是否正常

其他功能反复进行操作，查看系统资源是否正常

4) 性能评估：评估典型用户应用场景下，系统资源的使用情况。

可测试的方面- 安装和启动时间- CPU 的占用- 内存的占用- 流量的耗用- 电量的耗用- 后端, 测试 App 中的各类操作是否满足用户响应时间要求，主要是测试点在网速方面，2g, 3g, wifi, 4g 一定要覆盖到- 后端 有网络并发

/ 16 /

交叉事件测试

针对智能终端应用的服务等级划分方式及实时特性所提出的测试方法。交叉测试又叫事件或冲突测试，是指一个功能正在执行过程中，同时另外一个事件或操作对该过程进行干扰的测试。

如：App 在前/后台运行状态时与来电、文件下载、音乐收听等关键运用的交互情况测试等。交叉事件测试非常重要，能发现很多应用中潜在的性能问题。

- 1) 多个 App 同时运行是否影响正常功能
- 2) App 运行时前/后台切换是否影响正常功能
- 3) App 运行时拨打/接听电话 4) App 运行时发送/接收信息
- 5) App 运行时发送/收取邮件
- 6) App 运行时切换网络（2G、3G、4G、WIFI）
- 7) App 运行时浏览网络

- 8) App 运行时使用蓝牙传送/接收数据
- 9) App 运行时使用相机、计算器等手机自带设备

/ 17 /

兼容测试

主要测试内部和外部兼容性

- 1) 与本地及主流 App 是否兼容
- 2) 基于开发环境和生产环境的不同, 检验在各种网络连接下(WiFi、GSM、GPRS、EDGE、WCDMA、CDMA1x、CDMA2000、HSPDA 等), App 的数据和运用是否正确
- 3) 与各种设备是否兼容, 若有跨系统支持则需要检验是否在各系统下, 各种行为是否一致

不同操作系统的兼容性, 是否适配

不同手机屏幕分辨率的兼容性

不同手机品牌的兼容性

/ 18 /

回归测试

老徐备注: 不多说, 同其它测试一样;

/ 19 /

用户体验测试

以主观的普通消费者的角度去感知产品或服务的舒适、有用、易用、友好亲切程度。通过不同个体、独立空间和非经验的统计复用方式去有效评价产品的体验特性并提升产品的潜在客户满意度。

- 1) 是否有空数据界面设计, 引导用户去执行操作。

- 2) 是否滥用用户引导。
- 3) 是否有不可点击的效果，如：你的按钮此时处于不可用状态，那么一定要灰掉，或者拿掉按钮，否则会给用户误导
- 4) 菜单层次是否太深
- 5) 交互流程分支是否太多
- 6) 相关的选项是否离得很远
- 7) 一次是否载入太多的数据
- 8) 界面中按钮可点击范围是否适中
- 9) 标签页是否跟内容没有从属关系，当切换标签的时候，内容跟着切换
- 10) 操作应该有主次从属关系
- 11) 是否有横屏模式的设计，应用一般需要支持横屏模式，即自适应设计

老徐备注：如上简单列出，基本同其它测试差不多

/ 20 /

硬件环境测试

手势操作测试

- 1) 手机开锁屏对运行中的 App 的影响
- 2) 切换网络对运行中的 App 的影响
- 3) 运行中的 App 前后台切换的影响
- 4) 多个运行中的 App 的切换
- 5) App 运行时关机
- 6) App 运行时重启系统
- 7) App 运行时充电
- 8) App 运行时 kill 掉进程再打开

/ 21 /

网络环境

手机的网络目前主要分为 2G、3G、4G、wifi。

目前 2G 的网络相对于比较慢，测试时尤其要注意此块的测试。

- 1) 无网络时，执行需要网络的操作，给予友好提示，确保程序不出现 crash。
- 2) 内网测试时，要注意选择到外网操作时的异常情况处理。
- 3) 在网络信号不好时，检查功能状态是否正常，确保不因提交数据失败而造成 crash。
- 4) 在网络信号不好时，检查数据是否会一直处于提交中的状态，有无超时限制。如遇数据交换失败时要给予提示。
- 5) 在网络信号不好时，执行操作后，在回调没有完成的情况下，退出本页面或者执行其他操作的情况，有无异常情况。此问题也会经常出现程序 crash。

IDO 老徐，注：一般可以选择在地铁、公交、电梯、隧道灯代表性的场所进行测试。

或者通过 fiddler/charles 或其他工具，模拟弱网

/ 22 /

再次总结

1. UI 测试

app 主要核 ui 与实际设计的效果图是否一致；交互方面的问题建议，可以先与产品经理确认，确认通过后，才开始让开发实施更改或优化

2. 功能测试

根据软件说明或用户需求验证 App 的各个功能实现；

与其它测试一样，无太大区别，可以看看之前的测试点梳理文章，以及其它老徐的文章；

3. 中断测试

模拟用户真实使用 app 时会遇到的中断情况进行测试.

如：网络的断网，切换网络，断电，来电话/短信，听音乐，切换到其他 app，打开其他 app 的通知、低电量时能否正常使用等

4. 兼容以及适配测试

新旧版本的在功能，逻辑层面的兼容测试，同一个 app 在不同系统版本运行，以及不同机型之间的适配测试兼容测试：接口的兼容性测试能够保证大部分的功能完善；

app 在不同系统版本上保证运行适配性：屏幕，系统版本等（系统位数一定要考虑）该部分通过第三方的云平台进行

用不同的支持语言验证 App 行为。

5. 安全测试

App 安全测试大概划分为以下几类：

1) 从数据的本地存储到数据的传输、处理以及远程访问等各个环节，基于相应的安全标准/行业标准评估 App 的安全特性；

借鉴在 WebApp 和网络安全测试的一些成功经验在智能终端 App 测试中进行裁减或适配；

2) 检测 App 的用户授权级别，数据泄漏，非法授权访问等；

3) 对 App 的输入有效性校验、认证、授权、敏感数据存储、数据加密等方面进行检测, 以期发现潜在的安全问题；

4) 基于各种通信协议或相应的行业安全标准检视 App 是否满足相应的要求。

6. 线上测试

线上测试是产品上线之后一定要完成的，这部分可以根据场景化进行回归测试，其中网络环境要全部覆盖一遍

/ 23 /

测试 App 的消息显示和通知显示

在安装 App 的时候是否申请相应的权限

验证收到 App 新的消息的时候，如何通过通知向用户进行消息展示

测试 App 能否及时显示和同步消息

测试在多台设备登录 App，是否同步消息、设置、内容、数据等

测试 App 的消息是否同步在 PC 端（无论 Web 版本或者 PC 版本）

测试 App 中高内存使用的功能

测试 App 对于读取大量图片、视频等进行高内存占用操作的处理能力

查看 App 是否有对应的异步加载功能等

测试 App 的流量和电量消耗

测试 App 的安装文件大小

测试 App 占用的存储空间

测试 App 使用的流量和电量

测试 App 支持的文件格式

测试 App 支持 Office 文件操作

测试 App 支持图片文件展示

测试 App 支持 PDF 文件打开

测试 App 支持视频和音频文件播放

/ 24 /

需要考虑的一些情况：

- 1) 如果程序里面包括分享功能，那么检测点击分享的时候是否会正常给出分享提示，点击分享后所填写的分享内容是否正确
- 2) app 在清空数据或强制退出后还能正常运行否
- 3) api，包括在 app 内跳转到另一个界面，在返回来，以及跳转到系统 api
- 4) 长时间开机且开 app，看是否会出现异常情况

/ 25 /

最后，附上

移动 App 崩溃的一些常见原因：

设备碎片化：由于设备极具多样性，App 在不同的设备上可能有表现不同。

带宽限制：带宽不佳的网络对 App 所需的快速响应时间可能不够。

网络的变化：不同网络间的切换可能会影响 App 的稳定性。

内存管理：可用内存过低，或非授权的内存位置的使用可能会导致 App 失败。

用户过多：连接数量过多可能会导致 App 崩溃。

/ 26 /

关于性能测试，可以了解下 Jmeter 这款工具 。

以及

monkey / GT / MonkeyRunner

IDO 老徐，注：

具体工具怎么用，搜索引擎检索，大把资料。

附件

三张上线前检查清单

此份检查单，来源于网络，可参考

APP 上线前测试确认清单

APP 名称	
版本	
提交时间	

这个表格是用来记录每个 APP 版本提交到 App Store 前 App 的测试情况的。请对已经做过测试的项目后勾选“是”。如果没有做过测试则勾选“否”，如果该项对这个 APP 不适用，则勾选“N/A”。

连网情况	N/A	是	否
在下列连网状态下测试 app 的数据下载功能。请考虑优雅降级（Graceful degradation）的网络连接状况，网络连接状态失败，网络连接良好的情况			
WiFi			
Edge			
GPRS			
无网			
使用代理			
无法连接服务器			
断网之后恢复（如果有流媒体服务）			

本地设置	N/A	是	否
改变设备的设置然后加载 App，检查日期的显示是否正确，特别需要检查 app 推送的 feed 或者服务上的时间			
12 小时/24 小时时钟显示			
各个区域的格式：英国，美国，威尔士，法国，德国，澳大利亚，日本，俄国			
语言：英语，法语，德语，希腊语，日语			
时区			
GMT/BST			

设备	N/A	是	否
iOS 5.0 系统的 iPhone/iPod touch			
iOS 5.1.1 系统的 iPhone/iPod touch			
iOS 6.0 系统的 iPhone/iPod touch			
iOS 6.1.3 系统的 iPhone/iPod touch			
iOS 7.0 系统的 iPhone/iPod touch			
Retina 显示屏的 iPhone			
非 Retina 显示屏的 iPhone			
iOS 5.0 系统的 iPad 1			
iOS 5.1.1 系统的 iPad 1			

iOS 6.0 系统的 iPad			
iOS 6.1.3 系统的 iPad			
iOS 7.0 系统的 iPad			
Retina 显示屏的 iPad			
非 Retina 显示屏的 iPad			
iPad mini			

音频	N/A	是	否
如果 App 支持播放音频，测试一下内容。如果有流媒体服务的话，请检查之前的联网部分测试情况			
耳机/扬声器线路			
音频接口电路			
iPod touch 音频线路（考虑没有使用扬声器的情况）			
静音开关功能			
如有电话则暂停音频			
背景音乐（如果支持）：重放以及多任务栏控件			
当另一个 app 开始播放音乐时候停止播放			
耳机远程控制器			
多任务屏幕音频控件			

视频	N/A	是	否
如果有流媒体服务，请检查之前的联网部分测试情况			
重放开始之前用户取消视频播放			
在视频的播放期间停止播放			
完整播放视频			
退出全屏			
视频线路			
视频全屏到正常屏幕之间的转换			

地理位置	N/A	是	否
GPS			
WiFi 位置			
基站位置			
无法定位			
没有返回结果（例：无法联网）			
关闭了定位服务			
当另一个 app 开始播放音乐时候停止播放			
耳机远程控制器			
App 的定位服务被取消			

登陆	N/A	是	否
登陆到服务器			
登陆错误			

UI 界面	N/A	是	否
测试 app 的每个主界面			
状态栏高度翻倍（例：打电话的时候）			
音频接口电路			
iPod touch 音频线路（考虑没有使用扬声器的情况）			
设备方向变化			
设备方向上下颠倒			
关闭重力感应			
打开了 VoiceOver 功能			
锁屏之后恢复			
放大缩放功能打开			

Core Data	N/A	是	否
用户输入验证错误			
网页服务器输入验证错误			
使用有效的无效的数据文件测试数据移动			
回滚数据			

安装	N/A	是	否
重新安装			
从上个最新版本进行更新			
从老版本进行更新			
回滚			

文本	N/A	是	否
晃一晃取消操作			
文本选择（包括关闭文本选择的情况）			
复制/黏贴			
键盘隐藏时编辑			

Misc	N/A	是	否
运动状态下			
点对点模式下进行测试			
版本号更新			
上线发布是 Bundle identifier 是否正确			

想了解更多？

欢迎关注老徐的公众号

Tester & 职场人 & 5W人都在关注

软件测试从业者综合能力提升&择业加薪

第一时间，更新至此公众号



长按二维码/微信扫码 关注老徐

老徐私人微信957863300

有问题，随时留言

如上，扫码可以关注公号
第一时间查阅老徐的最新文章

End