**测试用例设计**

# 移动端测试用例设计总结

## 1.安装/卸载；

A.升级注意点：升级提示、取消更新/强制更新、后台更新（ios的自动更新）、跨版本升级、升级过程中异常情况的处理（取消、死机、重启、断电、内存不足、断网）、升级进度、不同网络下升级；  
B.第三方软件支持：itools、豌豆荚、91助手、华为助手、360、应用宝等；  
C.安装关注点：版本号、渠道号、数字签名（用抓包工具辅助查看）、安装成功后启动向导、安装过程中对意外情况的处理（取消、死机、重启、断电、内存不足、断网）、安装进度条、主要功能流程；  
D.卸载关注点：卸载过程中的意外情况处理（取消、死机、重启、断电、内存不足、断网）、卸载进度条；  
E.真机上安装、卸载、高版本覆盖安装、低版本覆盖安装、卸载后安装高版本；  
F.第三方软件协助安装、卸载、高版本覆盖安装、低版本覆盖安装、卸载后安装高版本；  
G.在线升级：

## 2.业务功能测试；

A.注册：用户名密码的输入（同文本框编辑“2-1）”）、用户名密码长度限制、注册后的页面提示（手机短信提示）、前台和后台数据一致、；  
B.登录：用户名密码的输入（同登录“1-1）”）、非法登录次数限制、多设备登录（MTOP现有原则，一个应用同时只允许一台设备登录）、禁用账号登录、登录成功信息、登录后有注销按钮、登录超时处理、登录过程断网处理、登录过程切换网络；  
C.注销：注销后新账号登录、取消注销；  
D.应用前后台切换：app前后台切换、锁屏解屏、电话中断后回到app、必须处理的提示框处理后回到app、杀掉进程后重新启动app、有数据交换的页面注意进行前后台切换以及锁屏解屏；  
E.免登录：登录后杀掉进程重新启动app、无网络、切换用户登录、密码更换、主动退出登录下次启动app、卸载重装、在线更新、覆盖安装、跨版本安装、；  
F.数据更新：手动或自动刷新、从后台切换到前台时数据更新、实时更新、定时更新、数据展示的处理逻辑（服务器获取、本地缓存）、更新异常处理（弱网、断网、服务器响应异常、数据为空）；  
G.定位、相机、语音、蓝牙等服务：已开启、未开启根据提示开启、未开启并拒绝开启；  
H.时间测试：修改手机时区；  
I.推送测试：推送消息内容、推送消息链接跳转、免打扰或拒绝接收；  
J..交叉事件测试：  
K.根据需求文档、原型图和设计稿验证app各个功能的实现；　　　  
L.共性功能：

## 3.兼容测试；

A.iphone、华为、小米、oppo、vivo、魅族等  
B.ios：ios10、ios9、ios8、ios7；android：android6.0、android5.0、android4.等；C.主流分辨率：10801920、7201280、8001280、25601440、 2040x1080等D.非主流：10801800等  
E.分辨率；  
F.主流系统版本；  
G.不同厂家定制；  
H.不同尺寸：6寸、5寸、5.5寸、5.7寸、4.7寸、4寸等

## 4.稳定性测试：

A.monkey结合友盟持续使用8小时以上统计crash率；  
B.各种事件打扰，如插拔数据线、电话打扰、收发短信、切换网络、浏览网络、使用蓝牙传送/接收数据、相机等；  
C.多个运行中app切换测试；

## 5.性能测试：

A.monkey结合性能测试工具监控cpu、内存、流量、耗电量，性能测试工具如anothermoniter、腾讯的GT；  
B.评估典型用户应用场景下，系统资源的使用情况；  
C.大数据测试（如需要读取用户通讯录的情况）；  
D.不同网络响应速度、服务器接口压力测试；  
E.与竞品的Benchmarking（基线测试）；

## 6.网络测试：

A.无网络测试；  
B.弱网测试；  
C.外网测试；

## 7.界面易用性测试：

A.是否有空数据界面设计，引导用户去执行操作。  
B.是否滥用用户引导。  
C.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  
D.菜单层次是否太深；  
E.交互流程分支是否太多；  
F.相关的选项是否离得很远；  
G.界面中按钮可点击范围是否适中；  
H.当切换标签的时候，内容跟着切换；  
I.是否定义Back的逻辑。涉及软硬件交互时，Back键应具体定义  
J.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；　  
K.在不同的页面是否有导航连接、导航与页面风格一致；  
L.是否需要搜索；  
M.图片质量、同一页面图片颜色不宜过多、同一页面标签风格统一；  
N.文案：输入框中说明文字、页面文字正确性、敏感词汇、敏感图片（设计专利、版权、隐私等）；  
O.android体验规范：长按弹出删除选项（一时想不起来还有哪些，平时还是要多做总结）；  
P.ios体验规范：左滑弹出删除选项、左右滑动可翻页　  
Q.符合android或ios体验规范；  
R.符合用户体验规范：

## 8.安全测试：

1. 输人的密码将不以明文形式进行显示
2. 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人的位置上
3. 不同的应用程序的密码长度必需至少在4一8 个数字长度之间
4. 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侧除它的临时文件, 文件可能遭受人侵者的袭击, 然后读取这些数据信息。
5. 当将敏感数据输人到应用程序时, 其不会被储存在设备中
6. 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
7. 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
8. 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, 更不能在安全警告显示前，利用显示误导信息欺骗用户，应用程序不应该模拟进行安全警告误导用户
9. 在数据删除之前，应用程序应当通知用户或者应用程序提供一个“取消”命令的操作
10. “ 取消”命令操作能够按照设计要求实现其功能
11. 应用程序应当能够处理当不允许应用软件连接到个人信息管理的情况
12. 当进行读或写用户信息操作时, 应用程序将会向用户发送一个操作错误的提示信息
13. 在没有用户明确许可的前提下不损坏删除个人信息管理应用程序中的任何内容
14. 应用程序读和写数据正确。
15. 应用程序应当有异常保护。
16. 如果数据库中重要的数据正要被重写, 应及时告知用户
17. 能合理地处理出现的错误
18. 意外情况下应提示用户
19. 能够在安装设备驱动程序上找到应用程序的相应图标
20. 安装路径应能指定
21. 没有用户的允许, 应用程序不能预先设定自动启动
22. 卸载是否安全, 其安装进去的文件是否全部卸载
23. 卸载用户使用过程中产生的文件是否有提示
24. 其修改的配置信息是否复原
25. 卸载是否影响其他软件的功能
26. 卸载应该移除所有的文件
27. 限制/允许使用手机功能接人互联网
28. 限制/允许使用手机发送接受信息功能
29. 限制/允许应用程序来注册自动启动应用程序
30. 限制或使用本地连接
31. 限制/允许使用手机拍照或录音
32. 限制/允许使用手机读取用户数据
33. 限制/允许使用手机写人用户数据
34. 软件权限：
35. 安装/卸载安全性：
36. 数据安全性：

# 测试用例

## 好处：

1）清晰思路、避免遗漏  
当系统功能多且复杂时，根据系统每个模块，拆分功能点，花点时间思考并整理成文档，尽可能结合功能与业务，模拟不同场景，从根本上避免了直接测试系统的“手忙脚乱”。  
　　  
2）明确测试进展  
参考测试用例，可以清晰看到哪些用例执行了，哪些用例没有执行，从中看到测试进行到哪一步，以及结合问题管理平台，直观地从测试角度，分析项目进展。  
　　  
3）系统模块缺陷率  
根据测试用例发现的问题，可以看到哪个功能模块出现的bug较多。

## 方法：

1）等价类划分：输入的子集中选择，假如输入要求输入数字1到10，那么输入4和7去验证；  
2）边界值：输入支持的最大值，假如一个文本输入框最大值为100，输入的内容超过101；  
3）因果图：判定表，通过因果关系判定；  
4）错误推测法：基于测试经验推测系统在什么样的操作可能会出现错误；

## 元素：

1）目录  
根据拆分的系统功能点，每个功能点可以用目录的方式区分，如一个系统的系统管理-用户管理-用户添加，那么系统管理就是一级目录，用户管理就是二级目录，用户添加就是三级目录；　　  
2）测试项  
同上，根据三级目录，一般用户添加包括的元素有用户名、密码输入框、保存按钮、取消，那么每个元素都可以分成一个测试项；　　  
3）操作步骤  
每一个测试项，都对应相应的操作步骤，可以分成操作步骤1、操作步骤2，操作步骤可以细化到，点击添加用户的按钮、输入框输入某某用户名密码点击保存按钮；　　  
4）预期结果  
操作步骤，在完成一系列动作之前，都要有对应预期结果作为参考，这个参考就是最初业务部门提供的需求分析文档，根据需求分析文档来告诉我们每一个功能要有什么样的结果；  
5）实际结果  
操作步骤，完成之后，需要记录实际的情况，如果与预期结果不符，就可以归于bug；  
考虑是否可以将实际结果改成注释，避免与预期结果一一对应关系，解耦合。　  
6）优先级  
探索性测试  
在设计测试用例时，并不能完全保证每个功能每个场景都设计到位，而且单纯执行测试的时候非常枯燥，那么加入一点发散性思维，进行一些非常规性操作，以用户的心态去使用系统，或是尽情地“破坏”系统，发现系统的问题。  
真正的探索性测试需要对产品的深入了解，以及软件开发技术有一定的深度和宽度。

## 用例评审

测试用例编写完成后，需要在测试内部进行一次评审，评审的内容包括：功能是否完整、需求是否符合，使每一个测试人员在系统测试过程中，不存在主体思维的偏差。  
同时用例评审也是一个很好的学习过程，每一位测试人在介绍系统时，能够意识到各自设计用例时的不足，包括自动化、性能都可以进行技术的探讨。

## 不需要用例情况：

1）功能过于简单  
2）急于交付，弊端就是不能保证测试的覆盖率。