

第一部分 背景

- 1、 为什么
提高产品的稳定性
提高产品的留存率
- 2、 什么时候开始
首轮功能测试通过后

第二部分 理论部分

- 1、 手工测试场景
- 2、 自动测试场景(模拟事件流)
TouchInput--KeyEvent—TouchSearch—TouchName—TouchFriend—
KeyEvent
- 3、 Monkey
 - 1) 什么是 Monkey
Monkey 是发送伪随机用户事件的工具
 - 2) Monkey 在哪里
在手机系统里
- 4、 ADB 命令
 - 1) 什么是 ADB ?
Android Debug Bridge
- 5、 MonkeyScript
 - 1) MonkeyScript 是一组可以被 Monkey 识别的命令集合
 - 2) MonkeyScript 可以完成重复固定的操作
- 6、 MonkeyRunner (手机截屏)
 - 1) 什么是 MonkeyRunner
提供了一系列的 API 和可以完成模拟事件及截图操作
 - 2) Monkey 和 MonkeyRunner 的区别
Monkey：在 adb shell 中，生成用户或者系统的伪随机事件
MonkeyRunner：通过 API 定义特定命令和事件控制设备
 - 3) MonkeyRunner APIs 分类
 - MonkeyRunner:用来连接设备或者模拟器
 - MonkeyDevice：提供安装、卸载应用，发送模拟事件
 - MonkeyImage：完成图像保存，及对比操作

- 4) MonkeyRunner 测试的类型
 - 多设备测试、功能测试、回归测试
- 7、 压力测试结果分析
 - 1) 什么是 CRASH ?
 - 即崩溃，应用程序在使用过程中，非正常退出
 - 2) 什么是 ANR (Application Not Responding)
 - Force close 、 Wait

第三部分 实践

准备环境：Android SDK 环境、Python 环境

- 1、 一个 App 的压测实战
 - 1) 在手机开发者选项中，将 USB 调试选上
 - 2) 确认电脑和手机已连接
 - `adb devices`
 - 3) 安装 App
 - `adb install package.apk`
 - 4) 发送压力指令
 - `adb shell monkey 1000`
 - `//activityResuming(com.android.settings)`说明是进入了后台操作
 - Events injected:1000 说明执行了 1000 次操作且没有异常
 - 5) 获取包名
 - `adb logcat | grep START`
 - `./uiautomatorviewer`
 - 6) 给指定的包打压力
 - `adb shell monkey -p com.android.calculator2 1000`
- 2、 Monkey 高级参数的应用
 - 1) throttle 参数
 - 指定事件之间的间隔
 - `sdb shell monkey --throttle <milliseconds>`
 - `adb shell monkey -p com.android.calculator2 --throttle 1000` (1000 毫秒) 100
 - 2) seed 参数
 - 指定随机生成数的 seed 值

```
adb shell monkey -s <seed> <event-count>
```

```
adb shell monkey -p com.android.calculator2 -s 100 50
```

3) 触摸事件

设定触摸事件的百分比

```
adb shell monkey -pct-touch <percent>
```

```
adb shell monkey -v -p com.android.calculator2 --pct-touch 100 (100%)  
100
```

Event percentages:查看各个事件

4) 动作事件

设定动作事件的百分比

```
adb shell monkey --pct-motion <percent>
```

```
adb shell monkey -v -p com.android.calculator2 --pct-touch 10 --  
pct-motion 10 (100%) 100
```

5) 轨迹球事件

设定轨迹球事件百分比

```
adb shell monkey --pct-trackball <percent>
```

6) 基本导航事件

设定基本导航事件的百分比，输入设备的上、下、左、右

```
adb shell monkey -pct-nav <percent>
```

7) 主要导航事件

设定主要导航事件百分比，兼容中间键、返回键、菜单按钮（软键盘）

```
adb shell monkey --pct-majornav <percent>
```

8) 系统导航事件

设定系统导航事件百分比，HOME、BACK、拨号及音量键

```
adb shell monkey --pct-syskeys <percent>
```

9) 启动 Activity 事件

设定启动 Activity 的事件比例

```
adb shell monkey --pct-appswitch <percent>
```

10) 不常用事件

设定不常用事件的百分比

```
adb shell monkey --pct-anyevent <percent>
```

11) 崩溃事件

忽略崩溃和异常

adb shell monkey --ignore-crashes <event_count>

12) 超时事件

忽略超时事件

adb shell monkey --ignore-timeouts <event-count>

adb shell monkey -v -p com.android.calculator2 --pct-touch 50 --pct-motion 50 --ignore-crashes --ignore-timeouts 100

3、 CRASH 结果的提取

Crash 之后，最后会有一个 seed 值，可以复现 bug

adb shell monkey -v -p com.android.calculator2 --pct-touch 10 --pct-motion 10 (100%) -s 343434 100

Ignore crash 之后不会再有 seed 值

4、 ANR 结果的提取

1) 使用 monkey 出现，会有 ANR 的提示

2) 手动的话，使用：

- adb shell 进入 shell 命令
- cd data/anr/ 进入 anr 目录下
- ll traces.txt 查看 traces 文件可以看到 anr 的信息

5、 Monkey Script 的实例

执行 Monkey 脚本的命令：

adb shell monkey -f <scriptfile> <event-count>

6、 MonkeyRunner 的实例

1) DispatchTrackball 命令，需配对使用

DispatchTrackball(long downtime, long eventide, int action, float x, float y, float pressure, float size, int metastate, float xprecision, float yprecision, int device, int edgeflags)

long downtime：键最初按下的时间

long eventide：事件开始的时间

action 0:按下 1：弹起

float yprecision:y 的精确值

int device：事件的来源

x,y 代表坐标点

2) DispatchPointer 命令

点击事件,需配对使用

DispatchPoiner(long downtime, long eventide , int action ,float x,
float y, float pressure, float size , int metastate , float xprecision ,
float yprecision, int device, int edgeflags)

3) DispatchString 命令

输入字符串命令

DispatchString(String text)

4) LanuchActivity 命令

启动应用

LanuchActivity(package, Activity)

5) UserWait 命令

等待事件

UserWait(1000)

6) DispatchPress 命令

按下键值

DispatchPress(int keycode) #keycode 66 回车键

实例：搜索关键词

- 1、 启动 app
- 2、 点击输入框
- 3、 输入查询词
- 4、 点击回车
- 5、 点击搜索按钮
- 6、 等待结果的出现
- 7、 点击 clear 按钮

● 命令行启动 uiautomatorviewer 的方法：./ uiautomatorviewer

7、MonkeyRunner (MonkeyScript 无法截屏)

分三个模块第一个是 MonkeyRunner

1) MonkeyRunner API –alert

警告框

void alert(string message, string title, string okTitle)

2) MonkeyRunner API –waitForConnection

等待设备连接， 有多个 device id,需要指明哪个设备

waitForConnection(float timeout, string deviceid)

3) MonkeyDevice API – drag

Drag(tuple start, tuple end, float duration,integer steps)

start:起点位置

end:终点的位置

duration:手势持续的时间

steps:差值点的步数，默认 10

4) MonkeyDevice API – press

按键

press(string keycode,dictionary type)

keycode 名, Down,UP,DOWN_AND_UP

5) MonkeyDevice API –startActivity

启动应用

startActivity(package+' /' +acvitivity)

6) MonkeyDevice API – touch

点击

touch(integer x, integer y, integer type)

x 坐标值, y 坐标值

type:DOWN,UP,DOWN_AND_UP

7) MonkeyDevice API – type

输入

type(string meaasge)

8)MonkeyDevice API – takeSnapshot

截屏

MonkeyImage takeSnapshot()

9)MonkeyImage API –sameAs

图像对比

boolean sameAs(MonkeyImage other,float percent)

10)MonkeyImage API – writetoFile

保存图像文件

void writeToFile(string path, string format)

- `./uiautomatorviewer`

- 使用 `monkeyrunner test.py`