

项目编号： KY2014485B

苏州大学大学生课外学术科研基金项目

结题报告

项 目 名 称： 基于 android 移动设备的 GUI 自动化测试研究

项 目 负 责 人： 葛 高 坚

项目指导教师： 张 广 泉

项目所在单位： 计算机科学与技术学院

项目起止时间： 2014 年 3 月——2014 年 9 月

1. 项目背景

随着信息产业技术的高速发展，计算机的应用领域越来越广泛，软件规模越来越庞大、软件结构越来越复杂，这就使保证软件的质量、高度可靠性面临巨大的挑战，例如美国航天局火星登陆事故，跨世纪“千年虫问题”等问题。2002 年，美国国家标准与技术研究所的一项研究表明，软件错误每年给美国所造成的损失 595 亿，该研究还证实如果经过测试及早发现错误有超过 1/3 的损失就可避免。虽然程序正确性证明等形式化方法能在一定程度上排除错误，但是这些方法都存在局限，远未达到广泛实用阶段。程序代码是软件质量的最终体现，软件测试在今后较长时间内仍将是保证软件质量的重要手段。

G. J. Myers 在其经典论著《The Art of Software Testing》中对软件测试提出如下观点：

- (1) 测试是程序的执行过程，目的在于发现错误；
- (2) 一个好的测试用例是可以发现至今尚未发现的错误；
- (3) 一个成功的测试是发现了至今未发现的错误。

1983 年，IEEE 对软件测试进行了准确的定义：软件测试是使用人工或自动手段来运行或测定某个系统的过程，检验它是否满足规定的需求或者弄清预期结果与实际结果之间的差别。

软件测试自动化，是现在软件测试领域的一个重要组成部分，就是执行某种程序设计语言编制的自动测试程序，控制被测软件的执行，模拟手动操作步骤，它可以代替大量的、重复的手工测试，可以降低人为的操作失误，减少测试工作量，提高测试效率，是回归测试的一个重要手段。随着信息技术的日益发展，软件业竞争的日趋激烈，人们对软件产品的质量意识逐渐加强。仅仅依靠以劳动密集型为特征的传统手工测试，已经不能满足人们对快节奏的软件开发和测试的需求。自动化测试技术可以大大减少测试开销，同时增加有限时间内的测试。

GUI 测试是确保图形界面应用程序 GUI 正确的行为和状态的过程。这包括验证数据处理，控制流，状态和显示窗口和对话框，例如[2]。在软件开发中，在一般情况下，一个适当的图形用户界面，以及其正确的行为和操作已经成为验收以图形用户界面为中心的应用程序成功的一个关键的因素。尤其是对于在有限的显示尺寸和特殊输入设备（如移动设备 手指，棍棒，或小键盘）等应用程序，这更加重要。随着移动应用技术快速发展和智能手机不断普及和发展，相对于桌面应用程序，移动软件应用程序越来越注重于好的用户体验。手机设备高分辨率的显示屏为用户提供了更好的交互式体验。确保 GUI 的正确行为已经成为成功的至关重要的一个因素。

然而，随着移动 GUI 应用程序更高级的 GUI 功能，测试面临着特殊挑战：GUI 的事件驱动性质使 GUI 应用程序不确定性增加，用户可以点击屏幕上的任何地方。对于移动应用开发

平台通常是来自不同目标平台。测试 GUI 所有的可能状态，这几乎是不现实的。这使得移动 GUI 测试比纯功能测试[3]更加困难。手动 GUI 测试是很容易出错，重复性差，导致工作量非常大。随着 android 移动系统越来越普及，以及相应的 android 应用程序也越来越多，以及手工测试在 GUI 中的固有限制，急需自动化 GUI 测试。自动化 GUI 测试可以使得移动应用程序情况明显改善，包括回归测试。GUI 自动化测试的思想是利用测试脚本来模拟用户与 GUI 应用程序的交互并验证在 GUI 中正确的行为，状态和控制流，从而发现可能偏离预期的行为。该测试结构化是从用户角度来阐明 GUI 应用程序需求分析。因此，构建基于 android 平台的 GUI 自动化测试对于提供高效的移动应用程序开发必不可少。

2. 研究思路

- 1) 首先了解了Android系统框架
- 2) 然后对GUI测试进行了相关的研究与分析
- 3) 接着对Android平台GUI测试方法，测试工具进行研究与分析
- 4) 使用Robotium对现有apk文件的测试
- 5) 最后编写简单的android 界面，并对源码进行GUI测试

3. 基于 Android 平台的测试工具比较，及 Robotium 介绍

Android APIs 提供的 instrumentation 类可以初始化 Android 应用程序代码，允许你监控应用程序的系统交互，配合 KeyEvent、MotionEvent 类，发送用户事件，进而实现 GUI 层的自动化。测试程序需要继承 ActivityInstrumentationTestCase2 来实现自动化。

为了方便编写自动化测试用例，我们需要对 ActivityInstrumentationTestCase2 进行扩展。业界也已经有一些成熟的自动化工具，诸如 Robotium、Athrunk、NativeDriver、MonkeyRunner 等。对于移动客户端 GUI 的自动化而言，需要保证选取的工具具有以下几点特性：

1. 工具开源，易于扩展。
2. 脚本编写简洁，维护成本低。
3. 满足客户端的自动化需求。
4. 便与校验结果的正确性。
5. 可用于持续集成。

表 1 Android 自动化工具对比

| | Robotium | Athrun | NativeDriver | MonkeyRunner |
|-------|----------|--------|--------------|--------------|
| 脚本语言 | Java | Java | Java | Python |
| 易扩展 | ● | ● | ● | ● |
| 易维护 | ● | ● | ● | ○ |
| 满足需求 | ○ | ● | ● | ○ |
| 易校验 | ● | ● | ● | ○ |
| 可持续集成 | ● | ● | ● | ○ |

- MonkeyRunner 通过编写 Python 脚本来实现自动化，结果的验证是通过截屏比对图片来实现，验证方式不够灵活，不建议采用。
- NativeDriver[2] 是 WebDriver 接口的一种实现，使用移动客户端原生 UI 而不是浏览器 UI（Selenium）的自动化测试工具。类似于 selenium RC 的方式来运行测试程序，对于熟悉 WebDriver 的用户，上手会很快。从表 1 可以看出该工具也可以满足我们的自动化需求，但在调研初期，该工具提供的接口较少，没法满足测试需求，而今的完善程度也已经很高了。没有使用该工具可以认为是历史原因。
- robotium 是 android 自带类 Instrumentation 的一个封装，方便测试人员直接调用封装好的接口，也就是说，实际上我们直接使用 Instrumentation 也能够进行自动化测试，但 robotium 可以简化我们的测试步骤，我们只需要调用某个 robotium 的 API，传几个参数，就等于我们在调用一部分的 Instrumentation 帮我们实现测试。
- Athrun 是由淘宝公司开发的，它的实现和 Robotium 类似，提供的接口也很多，并且支持 webview，也可实现持续集成。对于我们的产品而言，满足自动化的需求。

Robotium 出现得比较早，功能较成熟，而且比较活跃，经常有新版本出来。Athrun，最主要特点是支持 WebView 的测试，在实用性上，个人认为 Athrun 较实用，但它目前还不是特别完善，Robotium 是 android 测试中一个简单而又强大的工具。说它简单，是因为 Robotium 是基于 Android 测试框架 InstrumentationTestCase2 进行的 2 次封装，把一些基本操作又简化了一遍，它的文档只有一个 Solo 类，而且配置步骤是固定的几步。说它强大是因为，它在有源码是可以测试源码，进行白盒测试。没有源码时，只有一个应用程序的 apk 也可以对其进行黑盒测试。而且对程序上的每一个按钮，文本控件，图像，等都可以捕捉到，既可以使用按钮显示的名字进行点击，也可以使用坐标进行点击。因此，本项目采用的是 Robotium 工具。

4. Robotium 实现自动化测试

◆ 对 apk 的测试

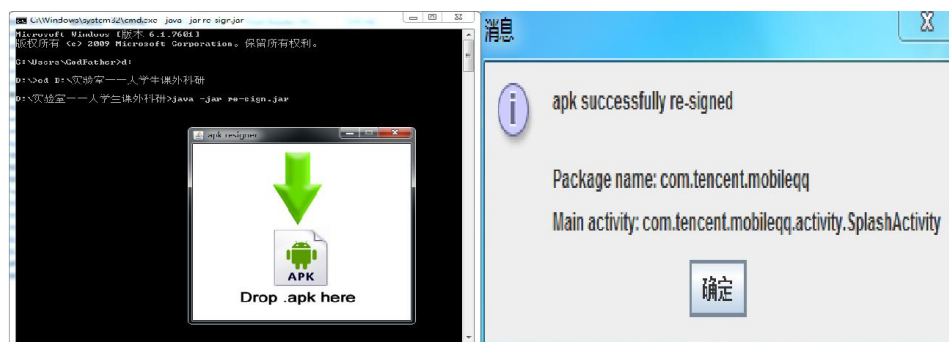
新建测试工程，及测试用例步骤如下：

Setp1: 在 eclipse 里面右键，点击“New”进入创建 Android 项目界面，选择“Android Test Project”，注意不是“Android Application Project”，点击“Next”，输入一个你的测试项目的名字。

Setp2: 没有源码时，选择“This Project”选项，点击“Next”，同样进入 android api 选择界面，选择好你要用到的 api 版本，点击“Finish”选项，即可完成项目的创建。

Setp3: 接着说导包的事情，进入 eclipse，选中你的项目，右键，点击“Properties”选项，再点击“Resource”选项下的“Java Build Path”，选中“Librararies”选项,添加 robotium 包，有时会出现这种情况：就是你进入“Java Build Path”选项后，会多出来一个“Android Dependencies”选项此时，你要做的就是，选中“Android Dependencies”，点击右侧的“Remove”选项，把这个去掉，否则你的测试程序会出现错误。

Setp4: 自己定义一个测试 case。利用 Robotium 编写 apk 自动化测试用例时，存在的两个问题：、1、如何找到要测试的 apk 的包名和主类名。2、怎样对要测试的 apk 进行去签名,并且使用 debug 签名。本项目使用 re-sign.jar 工具对将要测试的 apk 文件进行重签名，如图：



a)

b)

图 1 apk 重签名

Setp5 :将 apk 文件安装到虚拟环境中，进行测试，如图：

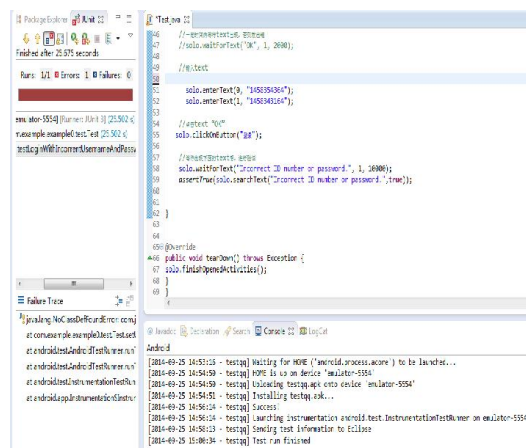


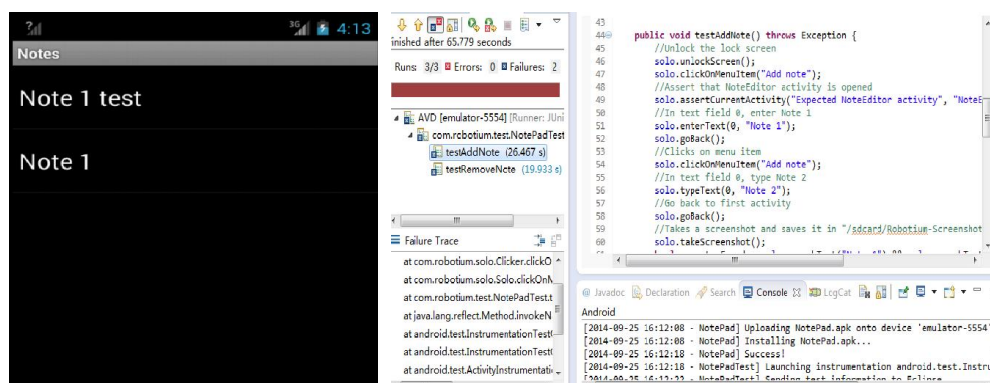
图 2 QQ apk 测试

◆ 对源码的测试

Setp1: 编写一个简单的记事本界面

Setp2: 程序源码导入到 eclipse 里面, 然后新建测试工程同上 Setp1

Setp3: 有源码时, 就点击 “An existing Android project” 选项, 然后选择你要测试的源码, 其它步骤同上, 在有源码时不需要上面的 Setp4、Setp5。



(a)

(b)

图 3 记事本源码测试

5. 项目总结

随着 android 移动系统越来越普及, 移动应用开发平台通常是来自不同目标平台, 测试 GUI 所有的可能状态, 这几乎是不现实的。本项目学习并构建了基于 Android 平台的 GUI 自动化测试, 可以大大减少测试开销, 同时增加有限时间内的测试。

主要参考文献:

- [1] G. J. Myers. The Art of Software Testing. Hoboken: John Wiley & Sons Press, 1979
- [2] Gerrard Paul, "Testing GUI Applications", EuroSTAR Conference, Edinburgh, November 1997, <http://www.gerrardconsulting.com/GUI/TestGui.html>, 18.07.2010
- [3] Brian Marick. When Should a Test Be Automated? < <http://www.exampler.com> >
- [4] Android Platform Development Kit, "Instrumentation Framework". Google 2008.
http://www.netmite.com/android/mydroid/development/pdk/docs/instrumentation_framework.html, 18.07.2010
- [5] Android Developers, Reference, "Android Test"
<http://developer.android.com/reference/android/test/package-summary.html>, 18.07.2010
- [6] 陈军成, 薛志成, 赵琛. 一种基于事件处理函数的 GUI 测试方法. 软件学报, 2013, 24(12): 2830-2842.