



(12) 发明专利申请

(10) 申请公布号 CN 104407980 A

(43) 申请公布日 2015. 03. 11

(21) 申请号 201410782404. 0

(22) 申请日 2014. 12. 17

(71) 申请人 用友软件股份有限公司
地址 100094 北京市海淀区北清路 68 号用
友软件园

(72) 发明人 刘建男

(74) 专利代理机构 北京中恒高博知识产权代理
有限公司 11249
代理人 宋敏

(51) Int. Cl.
G06F 11/36(2006. 01)

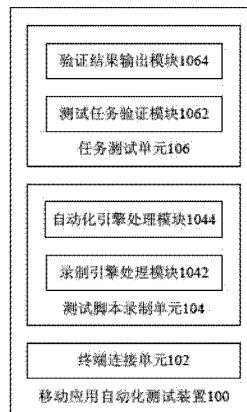
权利要求书3页 说明书8页 附图4页

(54) 发明名称

移动应用自动化测试装置和方法

(57) 摘要

本发明提供了一种移动应用自动化测试装置,包括:终端连接单元,用于在 PC 端安装待测应用程序,手机通过 USB 线连接 PC 端,在 USB 调试模式下识别到手机连接;测试脚本录制单元,用于利用 android 系统对待测应用程序进行录制、翻译和回放处理,和 / 或,利用 iOS 系统对待测应用程序进行录制和回放处理;任务测试单元,用于在自动化脚本中加入相应验证点,对待测应用程序进行验证,并输出验证结果。本发明还提供了一种移动应用自动化测试方法。通过本发明的技术方案,可以在现有的应用测试方式基础上,充分利用简单对象类型完成复杂对象类型的应用测试,建立多对象类型参与的应用测试的通用、统一测试思路。



1. 一种移动应用自动化测试装置,其特征在于,包括:

终端连接单元,用于在 PC 端安装待测应用程序,手机通过 USB 线连接 PC 端,在 USB 调试模式下识别到手机连接;

测试脚本录制单元,用于当识别到有手机连接时,利用 android 系统对待测应用程序进行录制、翻译和回放处理,和/或,利用 iOS 系统对待测应用程序进行录制和回放处理;

任务测试单元,用于基于对待测应用程序的处理结果,在自动化脚本中加入相应验证点,对待测应用程序进行验证,并输出验证结果。

2. 根据权利要求 1 所述的移动应用自动化测试装置,其特征在于,所述测试脚本录制单元,具体包括:

录制引擎处理模块,用于当识别到有手机连接时,利用 android 系统对待测应用程序进行录制、翻译和回放处理;

和/或,

自动化引擎处理模块,用于当识别到有手机连接时,利用 iOS 系统对待测应用程序进行录制和回放处理。

3. 根据权利要求 2 所述的移动应用自动化测试装置,其特征在于,所述录制引擎处理模块利用 android 系统对待测应用程序进行录制的操作,进一步包括:

使用 android 系统自带的 adb 命令 getevent,用户在手机上操作,动作以事件流的方式发送到 PC 端上,解析盖事件流获取用户在手机屏幕上的点击位置;和/或,使用 adb 命令截图功能,将手机屏幕截图发送到工具 PC 端的界面中,用户用鼠标点击界面中的手机截图,获取手机屏幕上的点击位置;同时配合使用 sendkey 命令向手机发指令,手机屏幕能够 and PC 端界面中的截图屏幕保持一致;

获取用户在手机上的行为动作和/或坐标信息后,使用 uiautomator 获得当前屏幕中包含所有对象的坐标、所占用屏幕面积的信息;

和/或,

所述录制引擎处理模块利用 android 系统对待测应用程序进行翻译的操作,进一步包括:

使用开源的 Robotium 框架作为回放引擎,使用 java 语言读取录制的脚本 xml 文件,将其内容转译成为 Robotium 框架所用的测试用例;

和/或,

所述录制引擎处理模块利用 android 系统对待测应用程序进行回放的操作,进一步包括:

预置回放环境,对待测应用程序的 apk 包进行解包重签名再打包,重新安装到手机上,使得翻译好的用例能够在手机上回放之前录制的动作;

和/或,

所述自动化引擎处理模块利用 iOS 系统对待测应用程序进行录制的操作,进一步包括:

利用 UI Automation 对 Developer 签名的应用程序或者运行在模拟器里的应用程序进行录制,取得用户的行为动作,以对象形式保存;

和/或,

所述自动化引擎处理模块利用 iOS 系统对待测应用程序进行回放的操作,进一步包括:

将待测应用程序进行 Developer 签名后,拿到源码后配置上 developer 的证书,直接通过 Xcode 编译安装到真实设备中,回放 UI Automation 的自动化测试脚本。

4. 根据权利要求 1-3 中任一项所述的移动应用自动化测试装置,其特征在于,所述任务测试单元,具体包括:

测试任务验证模块,用于基于对待测应用程序的处理结果,在自动化脚本中加入相应验证点,将验证点翻译成能被回放的验证语句,对待测应用程序进行验证;

验证结果输出模块,用于验证完成后,查看相应的回放结果,配置邮箱地址,将日志定时发送给相应人员。

5. 根据权利要求 4 所述的移动应用自动化测试装置,其特征在于,所述测试任务验证模块对待测应用程序进行验证的操作,进一步包括:

对待测应用程序的预期功能、正常条件和异常条件的验证;

和/或,

所述验证结果输出模块查看的回放结果,包括执行用例总数、通过数、失败数、跳过数、失败错误堆栈信息、失败截图。

6. 一种移动应用自动化测试方法,其特征在于,包括:

步骤 202:在 PC 端安装待测应用程序,手机通过 USB 线连接 PC 端,在 USB 调试模式下识别到手机连接;

步骤 204:当识别到有手机连接时,利用 android 系统对待测应用程序进行录制、翻译和回放处理,和/或,利用 iOS 系统对待测应用程序进行录制和回放处理;

步骤 206:基于对待测应用程序的处理结果,在自动化脚本中加入相应验证点,对待测应用程序进行验证,并输出验证结果。

7. 根据权利要求 6 所述的移动应用自动化测试方法,其特征在于,所述步骤 204,具体包括:

步骤 302:当识别到有手机连接时,利用 android 系统对待测应用程序进行录制、翻译和回放处理;

和/或,

步骤 304:当识别到有手机连接时,利用 iOS 系统对待测应用程序进行录制和回放处理。

8. 根据权利要求 7 所述的移动应用自动化测试方法,其特征在于,所述步骤 302 利用 android 系统对待测应用程序进行录制的操作,进一步包括:

使用 android 系统自带的 adb 命令 getevent,用户在手机上操作,动作以事件流的方式发送到 PC 端上,解析盖事件流获取用户在手机屏幕上的点击位置;和/或,使用 adb 命令截图功能,将手机屏幕截图发送到工具 PC 端的界面中,用户用鼠标点击界面中的手机截图,获取手机屏幕上的点击位置;同时配合使用 sendkey 命令向手机发指令,手机屏幕能够和 PC 端界面中的截图屏幕保持一致;

获取用户在手机上的行为动作和/或坐标信息后,使用 uiautomator 获得当前屏幕中包含所有对象的坐标、所占用屏幕面积的信息;

和 / 或,

所述步骤 302 利用 android 系统对待测应用程序进行翻译的操作,进一步包括:

使用开源的 Robotium 框架作为回放引擎,使用 java 语言读取录制的脚本 xml 文件,将其内容转译成为 Robotium 框架所用的测试用例;

和 / 或,

所述步骤 302 利用 android 系统对待测应用程序进行回放的操作,进一步包括:

预置回放环境,对待测应用程序的 apk 包进行解包重签名再打包,重新安装到手机上,使得翻译好的用例能够在手机上回放之前录制的动作;

和 / 或,

所述步骤 304 利用 iOS 系统对待测应用程序进行录制的操作,进一步包括:

利用 UI Automation 对 Developer 签名的应用程序或者运行在模拟器里的应用程序进行录制,取得用户的行为动作,以对象形式保存;

和 / 或,

所述步骤 304 利用 iOS 系统对待测应用程序进行回放的操作,进一步包括:

将待测应用程序进行 Developer 签名后,拿到源码后配置上 developer 的证书,直接通过 Xcode 编译安装到真实设备中,回放 UI Automation 的自动化测试脚本。

9. 根据权利要求 6-8 中任一项所述的移动应用自动化测试方法,其特征在于,所述步骤 206,具体包括:

步骤 402 基于对待测应用程序的处理结果,在自动化脚本中加入相应验证点,将验证点翻译成能被回放的验证语句,对待测应用程序进行验证;

步骤 404 验证完成后,查看相应的回放结果,配置邮箱地址,将日志定时发送给相应人员。

10. 根据权利要求 9 所述的移动应用自动化测试方法,其特征在于,所述步骤 402 对待测应用程序进行验证的操作,进一步包括:

对待测应用程序的预期功能、正常条件和异常条件的验证;

和 / 或,

所述步骤 404 查看的回放结果,包括执行用例总数、通过数、失败数、跳过数、失败错误堆栈信息、失败截图。

移动应用自动化测试装置和方法

[0001]

技术领域

[0002] 本发明涉及数据处理技术领域,具体地,涉及一种移动应用自动化测试装置和一种移动应用自动化测试方法。

[0003]

背景技术

[0004] 软件的自动化测试是把以人为驱动测试行为转化为机器执行的一种过程。测试人员按照测试用例编写自动化测试脚本,再由机器执行验证预先设置好的正常条件和异常条件。自动化脚本便可以在工程师们睡觉的时候运行,并且记录测试结果;自动化测试还可以不断地重复某个动作而不会感到疲倦;在添加了新功能以后,还可以回归一些原有功能的测试以保证其不受影响。在此过程中节省了人力、时间和硬件资源,提高测试效率和准确率。

[0005] 由于智能手机操作系统及 UI 与电脑相比的特殊性,为智能手机应用程序编写自动化测试脚本是一件非常复杂的事情,而且很难保证正确性。

[0006] 目前智能手机几乎已经普及到人手一部的程度,移动应用自动化测试是该领域不可回避的一个课题。

[0007] 因此,需要一种新的应用测试技术,可以在现有的应用测试方式基础上,充分利用简单对象类型完成复杂对象类型的应用测试,建立多对象类型参与的应用测试的通用、统一测试思路。

[0008]

发明内容

[0009] 本发明正是基于上述问题,提出了一种新的应用测试技术,可以在现有的应用测试方式基础上,充分利用简单对象类型完成复杂对象类型的应用测试,建立多对象类型参与的应用测试的通用、统一测试思路。

[0010] 有鉴于此,本发明提出了一种移动应用自动化测试装置,包括:终端连接单元,用于在 PC 端安装待测应用程序,手机通过 USB 线连接 PC 端,在 USB 调试模式下识别到手机连接;测试脚本录制单元,用于当识别到有手机连接时,利用 android 系统对待测应用程序进行录制、翻译和回放处理,和/或,利用 iOS 系统对待测应用程序进行录制和回放处理;任务测试单元,用于基于对待测应用程序的处理结果,在自动化脚本中加入相应验证点,对待测应用程序进行验证,并输出验证结果。在该技术方案中,可以方便地生成自动化脚本,使得测试人员在编写脚本方面的投入大大降低。

[0011] 在上述技术方案中,优选地,所述测试脚本录制单元,具体包括:录制引擎处理模块,用于当识别到有手机连接时,利用 android 系统对待测应用程序进行录制、翻译和回放

处理 ;和 / 或, 自动化引擎处理模块, 用于当识别到有手机连接时, 利用 iOS 系统对待测应用程序进行录制和回放处理。在该技术方案中, 可以被持续集成调用, 在自动编译、做盘后进行黑盒测试, 测试结果再持续集成系统读出展现, 使开发全流程实现闭环。

[0012] 在上述技术方案中, 优选地, 所述录制引擎处理模块利用 android 系统对待测应用程序进行录制的操作, 进一步包括 : 使用 android 系统自带的 adb 命令 `getevent`, 用户在手机上操作, 动作以事件流的方式发送到 PC 端上, 解析盖事件流获取用户在手机屏幕上的点击位置 ; 和 / 或, 使用 adb 命令截图功能, 将手机屏幕截图发送到工具 PC 端的界面中, 用户用鼠标点击界面中的手机截图, 获取手机屏幕上的点击位置 ; 同时配合使用 `sendkey` 命令向手机发指令, 手机屏幕能够和 PC 端界面中的截图屏幕保持一致 ; 获取用户在手机上的行为动作和 / 或坐标信息后, 使用 `uiautomator` 获得当前屏幕中包含所有对象的坐标、所占屏幕面积的信息 ; 和 / 或, 所述录制引擎处理模块利用 android 系统对待测应用程序进行翻译的操作, 进一步包括 : 使用开源的 Robotium 框架作为回放引擎, 使用 java 语言读取录制的脚本 xml 文件, 将其内容转译成为 Robotium 框架所用的测试用例 ; 和 / 或, 所述录制引擎处理模块利用 android 系统对待测应用程序进行回放的操作, 进一步包括 : 预置回放环境, 对待测应用程序的 apk 包进行解包重签名再打包, 重新安装到手机上, 使得翻译好的用例能够在手机上回放之前录制的动作 ; 和 / 或, 所述自动化引擎处理模块利用 iOS 系统对待测应用程序进行录制的操作, 进一步包括 : 利用 UI Automation 对 Developer 签名的应用程序或者运行在模拟器里的应用程序进行录制, 取得用户的行为动作, 以对象形式保存 ; 和 / 或, 所述自动化引擎处理模块利用 iOS 系统对待测应用程序进行回放的操作, 进一步包括 : 将待测应用程序进行 Developer 签名后, 拿到源码后配置上 developer 的证书, 直接通过 Xcode 编译安装到真实设备中, 回放 UI Automation 的自动化测试脚本。在该技术方案中, 可以提高测试效率, 使测试人员更加专注于新的测试模块的建立和开发, 从而提高测试覆盖率。

[0013] 在上述技术方案中, 优选地, 所述任务测试单元, 具体包括 : 测试任务验证模块, 用于基于对待测应用程序的处理结果, 在自动化脚本中加入相应验证点, 将验证点翻译成能被回放的验证语句, 对待测应用程序进行验证 ; 验证结果输出模块, 用于验证完成后, 查看相应的回放结果, 配置邮箱地址, 将日志定时发送给相应人员。在该技术方案中, 可以便于测试资产的数字化管理, 使得测试资产在整个测试生命周期内可以得到复用。

[0014] 在上述技术方案中, 优选地, 所述测试任务验证模块对待测应用程序进行验证的操作, 进一步包括 : 对待测应用程序的预期功能、正常条件和异常条件的验证 ; 和 / 或, 所述验证结果输出模块查看的回放结果, 包括执行用例总数、通过数、失败数、跳过数、失败错误堆栈信息、失败截图。在该技术方案中, 可以使机构的测试活动开展更加过程化, 符合 CMMI 过程改进的思想。

[0015] 根据本发明的又一个方面, 还提出了一种移动应用自动化测试方法, 包括 : 步骤 202 : 在 PC 端安装待测应用程序, 手机通过 USB 线连接 PC 端, 在 USB 调试模式下识别到手机连接 ; 步骤 204 : 当识别到有手机连接时, 利用 android 系统对待测应用程序进行录制、翻译和回放处理, 和 / 或, 利用 iOS 系统对待测应用程序进行录制和回放处理 ; 步骤 206 : 基于对待测应用程序的处理结果, 在自动化脚本中加入相应验证点, 对待测应用程序进行验证, 并输出验证结果。在该技术方案中, 可以方便地生成自动化脚本, 使得测试人员在编写脚本方

面的投入大大降低。

[0016] 在上述技术方案中,优选地,所述步骤 204,具体包括:步骤 302:当识别到有手机连接时,利用 android 系统对待测应用程序进行录制、翻译和回放处理;和/或,步骤 304:当识别到有手机连接时,利用 iOS 系统对待测应用程序进行录制和回放处理。在该技术方案中,可以被持续集成调用,在自动编译、做盘后进行黑盒测试,测试结果再持续集成系统读出展现,使开发全流程实现闭环。

[0017] 在上述技术方案中,优选地,所述步骤 302 利用 android 系统对待测应用程序进行录制的操作,进一步包括:使用 android 系统自带的 adb 命令 `getevent`,用户在手机上操作,动作以事件流的方式发送到 PC 端上,解析盖事件流获取用户在手机屏幕上的点击位置;和/或,使用 adb 命令截图功能,将手机屏幕截图发送到工具 PC 端的界面中,用户用鼠标点击界面中的手机截图,获取手机屏幕上的点击位置;同时配合使用 `sendkey` 命令向手机发指令,手机屏幕能够和 PC 端界面中的截图屏幕保持一致;获取用户在手机上的行为动作和/或坐标信息后,使用 `uiautomator` 获得当前屏幕中包含所有对象的坐标、所占用屏幕面积的信息;和/或,所述步骤 302 利用 android 系统对待测应用程序进行翻译的操作,进一步包括:使用开源的 Robotium 框架作为回放引擎,使用 java 语言读取录制的脚本 xml 文件,将其内容转译成为 Robotium 框架所用的测试用例;和/或,所述步骤 302 利用 android 系统对待测应用程序进行回放的操作,进一步包括:预置回放环境,对待测应用程序的 apk 包进行解包重签名再打包,重新安装到手机上,使得翻译好的用例能够在手机上回放之前录制的动作;和/或,所述步骤 304 利用 iOS 系统对待测应用程序进行录制的操作,进一步包括:利用 UI Automation 对 Developer 签名的应用程序或者运行在模拟器里的应用程序进行录制,取得用户的行为动作,以对象形式保存;和/或,所述步骤 304 利用 iOS 系统对待测应用程序进行回放的操作,进一步包括:将待测应用程序进行 Developer 签名后,拿到源码后配置上 developer 的证书,直接通过 Xcode 编译安装到真实设备中,回放 UI Automation 的自动化测试脚本。在该技术方案中,可以提高测试效率,使测试人员更加专注于新的测试模块的建立和开发,从而提高测试覆盖率。

[0018] 在上述技术方案中,优选地,所述步骤 206,具体包括:步骤 402 基于对待测应用程序的处理结果,在自动化脚本中加入相应验证点,将验证点翻译成能被回放的验证语句,对待测应用程序进行验证;步骤 404 验证完成后,查看相应的回放结果,配置邮箱地址,将日志定时发送给相应人员。在该技术方案中,可以便于测试资产的数字化管理,使得测试资产在整个测试生命周期内可以得到复用。

[0019] 在上述技术方案中,优选地,所述步骤 402 对待测应用程序进行验证的操作,进一步包括:对待测应用程序的预期功能、正常条件和异常条件的验证;和/或,所述步骤 404 查看的回放结果,包括执行用例总数、通过数、失败数、跳过数、失败错误堆栈信息、失败截图。在该技术方案中,可以使机构的测试活动开展更加过程化,符合 CMMI 过程改进的思想。

[0020] 通过以上技术方案,可以在现有的应用测试方式基础上,充分利用简单对象类型完成复杂对象类型的应用测试,建立多对象类型参与的应用测试的通用、统一测试思路。

[0021]

附图说明

[0022] 图 1 示出了根据本发明的实施例的移动应用自动化测试装置的框图；
图 2 示出了根据本发明的实施例的移动应用自动化测试方法的流程图；
图 3 示出了根据本发明的实施例的测试脚本录制单元的流程图；
图 4 示出了根据本发明的实施例的任务测试单元的流程图；
图 5 示出了根据本发明的实施例的工具示意图；
图 6 示出了根据本发明的实施例的业务流程图。

[0023]

具体实施方式

[0024] 为了能够更清楚地理解本发明的上述目的、特征和优点，下面结合附图和具体实施方式对本发明进行进一步的详细描述。需要说明的是，在不冲突的情况下，本申请的实施例及实施例中的特征可以相互组合。

[0025] 在下面的描述中阐述了很多具体细节以便于充分理解本发明，但是，本发明还可以采用其他不同于在此描述的方式来实施，因此，本发明的保护范围并不受下面公开的具体实施例的限制。

[0026] 图 1 示出了根据本发明的实施例的移动应用自动化测试装置的框图。

[0027] 如图 1 所示，根据本发明的实施例的移动应用自动化测试装置 100，包括：终端连接单元 102，用于在 PC 端安装待测应用程序，手机通过 USB 线连接 PC 端，在 USB 调试模式下识别到手机连接；测试脚本录制单元 104，用于当识别到有手机连接时，利用 android 系统对待测应用程序进行录制、翻译和回放处理，和 / 或，利用 iOS 系统对待测应用程序进行录制和回放处理；任务测试单元 106，用于基于对待测应用程序的处理结果，在自动化脚本中加入相应验证点，对待测应用程序进行验证，并输出验证结果。在该技术方案中，可以方便地生成自动化脚本，使得测试人员在编写脚本方面的投入大大降低。

[0028] 在上述技术方案中，优选地，测试脚本录制单元 104，具体包括：录制引擎处理模块 1042，用于当识别到有手机连接时，利用 android 系统对待测应用程序进行录制、翻译和回放处理；和 / 或，自动化引擎处理模块 1044，用于当识别到有手机连接时，利用 iOS 系统对待测应用程序进行录制和回放处理。在该技术方案中，可以被持续集成调用，在自动编译、做盘后进行黑盒测试，测试结果再持续集成系统读出展现，使开发全流程实现闭环。

[0029] 在上述技术方案中，优选地，录制引擎处理模块 1042 利用 android 系统对待测应用程序进行录制的操作，进一步包括：使用 android 系统自带的 adb 命令 getevent，用户在手机上操作，动作以事件流的方式发送到 PC 端上，解析盖事件流获取用户在手机屏幕上的点击位置；和 / 或，使用 adb 命令截图功能，将手机屏幕截图发送到工具 PC 端的界面中，用户用鼠标点击界面中的手机截图，获取手机屏幕上的点击位置；同时配合使用 sendkey 命令向手机发指令，手机屏幕能够和 PC 端界面中的截图屏幕保持一致；获取用户在手机上的行为动作和 / 或坐标信息后，使用 uiautomator 获得当前屏幕中包含所有对象的坐标、所占屏幕面积的信息；和 / 或，录制引擎处理模块 1042 利用 android 系统对待测应用程序进行翻译的操作，进一步包括：使用开源的 Robotium 框架作为回放引擎，使用 java 语言读取录制的脚本 xml 文件，将其内容转译成为 Robotium 框架所用的测试用例；和 / 或，录制引擎处理模块 1042 利用 android 系统对待测应用程序进行回放的的操作，进一步包括：预置回放

环境,对待测应用程序的 apk 包进行解包重签名再打包,重新安装到手机上,使得翻译好的用例能够在手机上回放之前录制的动作;和/或,自动化引擎处理模块 1044 利用 iOS 系统对待测应用程序进行录制的操作,进一步包括:利用 UI Automation 对 Developer 签名的应用程序或者运行在模拟器里的应用程序进行录制,取得用户的行为动作,以对象形式保存;和/或,自动化引擎处理模块 1044 利用 iOS 系统对待测应用程序进行回放的操作,进一步包括:将待测应用程序进行 Developer 签名后,拿到源码后配置上 developer 的证书,直接通过 Xcode 编译安装到真实设备中,回放 UI Automation 的自动化测试脚本。在该技术方案中,可以提高测试效率,使测试人员更加专注于新的测试模块的建立和开发,从而提高测试覆盖率。

[0030] 在上述技术方案中,优选地,任务测试单元 106,具体包括:测试任务验证模块 1062,用于基于对待测应用程序的处理结果,在自动化脚本中加入相应验证点,将验证点翻译成能被回放的验证语句,对待测应用程序进行验证;验证结果输出模块 1064,用于验证完成后,查看相应的回放结果,配置邮箱地址,将日志定时发送给相应人员。在该技术方案中,可以便于测试资产的数字化管理,使得测试资产在整个测试生命周期内可以得到复用。

[0031] 在上述技术方案中,优选地,测试任务验证模块 1062 对待测应用程序进行验证的操作,进一步包括:对待测应用程序的预期功能、正常条件和异常条件的验证;和/或,验证结果输出模块 1064 查看的回放结果,包括执行用例总数、通过数、失败数、跳过数、失败错误堆栈信息、失败截图。在该技术方案中,可以使机构的测试活动开展更加过程化,符合 CMMI 过程改进的思想。

[0032] 图 2 示出了根据本发明的实施例的移动应用自动化测试方法的流程图。

[0033] 如图 2 所示,根据本发明的实施例的移动应用自动化测试方法,包括:步骤 202:在 PC 端安装待测应用程序,手机通过 USB 线连接 PC 端,在 USB 调试模式下识别到手机连接;步骤 204:当识别到有手机连接时,利用 android 系统对待测应用程序进行录制、翻译和回放处理,和/或,利用 iOS 系统对待测应用程序进行录制和回放处理;步骤 206:基于对待测应用程序的处理结果,在自动化脚本中加入相应验证点,对待测应用程序进行验证,并输出验证结果。在该技术方案中,可以方便地生成自动化脚本,使得测试人员在编写脚本方面的投入大大降低。

[0034] 在上述技术方案中,优选地,如图 3 所示,步骤 204,具体包括:步骤 302:当识别到有手机连接时,利用 android 系统对待测应用程序进行录制、翻译和回放处理;和/或,步骤 304:当识别到有手机连接时,利用 iOS 系统对待测应用程序进行录制和回放处理。在该技术方案中,可以被持续集成调用,在自动编译、做盘后进行黑盒测试,测试结果再持续集成系统读出展现,使开发全流程实现闭环。

[0035] 在上述技术方案中,优选地,步骤 302 利用 android 系统对待测应用程序进行录制的操作,进一步包括:使用 android 系统自带的 adb 命令 `getevent`,用户在手机上操作,动作以事件流的方式发送到 PC 端上,解析盖事件流获取用户在手机屏幕上的点击位置;和/或,使用 adb 命令截图功能,将手机屏幕截图发送到工具 PC 端的界面中,用户用鼠标点击界面中的手机截图,获取手机屏幕上的点击位置;同时配合使用 `sendkey` 命令向手机发指令,手机屏幕能够和 PC 端界面中的截图屏幕保持一致;获取用户在手机上的行为动作和/或坐标信息后,使用 `uiautomator` 获得当前屏幕中包含所有对象的坐标、所占用屏幕面积的信

息 ;和 / 或,步骤 302 利用 android 系统对待测应用程序进行翻译的操作,进一步包括 :使用开源的 Robotium 框架作为回放引擎,使用 java 语言读取录制的脚本 xml 文件,将其内容翻译成为 Robotium 框架所用的测试用例 ;和 / 或,步骤 302 利用 android 系统对待测应用程序进行回放的的操作,进一步包括 :预置回放环境,对待测应用程序的 apk 包进行解包重签名再打包,重新安装到手机上,使得翻译好的用例能够在手机上回放之前录制的动作 ;和 / 或,步骤 304 利用 iOS 系统对待测应用程序进行录制的操作,进一步包括 :利用 UI Automation 对 Developer 签名的应用程序或者运行在模拟器里的应用程序进行录制,取得用户的行为动作,以对象形式保存 ;和 / 或,步骤 304 利用 iOS 系统对待测应用程序进行回放的的操作,进一步包括 :将待测应用程序进行 Developer 签名后,拿到源码后配置上 developer 的证书,直接通过 Xcode 编译安装到真实设备中,回放 UI Automation 的自动化测试脚本。在该技术方案中,可以提高测试效率,使测试人员更加专注于新的测试模块的建立和开发,从而提高测试覆盖率。

[0036] 在上述技术方案中,优选地,如图 4 所示,步骤 206,具体包括 :步骤 402 基于对待测应用程序的处理结果,在自动化脚本中加入相应验证点,将验证点翻译成能被回放的验证语句,对待测应用程序进行验证 ;步骤 404 验证完成后,查看相应的回放结果,配置邮箱地址,将日志定时发送给相应人员。在该技术方案中,可以便于测试资产的数字化管理,使得测试资产在整个测试生命周期内可以得到复用。

[0037] 在上述技术方案中,优选地,步骤 402 对待测应用程序进行验证的操作,进一步包括 :对待测应用程序的预期功能、正常条件和异常条件的验证 ;和 / 或,步骤 404 查看的回放结果,包括执行用例总数、通过数、失败数、跳过数、失败错误堆栈信息、失败截图。在该技术方案中,可以使机构的测试活动开展更加过程化,符合 CMMI 过程改进的思想。

[0038] 本发明的技术方案,正是为了解决现有技术中主流智能手机上应用程序自动化测试问题的工具 ;鉴于智能手机操作系统绝大部分被 android 和 iOS 占据,本发明的技术方案对 android 和 iOS 两个系统提供支持。

[0039] 本发明的技术方案,提供了一种移动应用自动化测试工具,在 android、iOS 主流智能手机操作系统上实现对其应用的自动化测试。本发明的技术方案需要解决的技术问题如下 :

1、自动化脚本生成。以录制的方式生成脚本,代替以往人工编写脚本的方式。

[0040] 2、脚本可移植性和复用性。工具录制出的自动化脚本以静态文本保存,文件本身不依赖于工具。

[0041] 3、脚本对回放环境的支持。自动化测试脚本中以对象的形式保存行为动作。只要被测程序中对象存在,而无须考虑回放设备的分辨率、设置等因素即可回放验证。

[0042] 4、回放脚本与回放框架相结合。选用性能优异的回放框架对脚本进行回放,节省系统资源。

[0043] 解决本发明技术问题的技术方案 :

(一)工具示意图。参见图 5。

[0044] (二)工具的 UI 及操作。

[0045] 该工具使用 java 语言编写,界面通过 swt 生成,简洁易懂。其操作流程是,在 PC 端安装并打开工具,手机通过 USB 线连接电脑并开打 USB 调试模式,这时工具会识别有手机

连接(iOS 系统手机连接时需要 PC 端也是苹果的操作系统)。再打开被测应用程序,按录制用例一翻译用例一回放用例的步骤进行。

[0046] (三)录制用例

1. android 系统录制用例：

在 android 系统中,有两种采集用户行为动作的方式,第一种方式使用 android 系统自带的 adb 命令 `getevent`,用户在手机上操作,动作会以事件流的方式发送到电脑上,解析这个事件流即可得到用户在手机屏幕上的点击位置。第二种方式使用 adb 命令截图功能,将手机屏幕截图发送到工具 PC 端的界面中,用户用鼠标点击界面中的手机截图,工具即可获得手机屏幕上的点击位置。同时配合使用 `sendkey` 命令向手机发指令,手机屏幕便可以和 PC 端界面中的截图屏幕保持一致。

[0047] 现在已经获得了用户在手机上的行为动作,也就是一些坐标信息。如 `click(200, 105)`、`slip(100,600,230,100)` 这样动作和坐标的组合,但这样的信息在不同分辨率的手机上回放会有问题。使用 `uiautomator` 可以获得当前屏幕中所有对象的坐标、所占用屏幕面积等信息,这样就可以把 `click(200,105)` 转化为 `click(objectName)`。

[0048] 2. iOS 系统录制用例：

UI Automation 可以对 Developer 签名的应用程序或者运行在模拟器里的应用程序进行录制。该工具封装了 UI Automation,利用其录制功能可以取得用户的行为动作。用例动作如下：

```
Var target = UITarget.localTarget();
```

这样的脚本也是以对象形式保存的,这和我们工具的设计初衷保持一致。

[0049] (四)翻译用例

1. android 系统使用开源的 Robotium 框架作为回放引擎。翻译时使用 java 语言读取录制的脚本 xml 文件,将其内容转译成为 Robotium 框架所用的测试用例。因为手机上操作种类有限,我们只需要关心诸如点击、双击、放大、缩小、滑动、拖动等动作即可。

[0050] 2. iOS 系统的用例不需要再次翻译,因为 UI Automation 录制出的脚本已经是可用的测试用例了。

[0051] (五)回放用例

1. android 系统中回放：

用例的回放动作会在手机屏幕上显示,但在回放执行前,需要预置工具预置回放环境。对于 android 系统的应用程序而言,只有重签名后才能被 Robotium 框架所认识,所以该工具要对被测程序的 apk 包进行解包重签名再打包,重新安装到手机上,这样翻译好的用例就可以在手机上回放之前录制的动作。

[0052] 2. iOS 系统中回放：

被测程序同样需要 Developer 签名,我们拿到源码后配置上 developer 的证书,直接通过 Xcode 编译安装到真实设备中,就可以回放 UI Automation 的自动化测试脚本了。

[0053] (六)结果验证

记用户的行为动作重现并不是最终目的,我们要实现对程序功能的验证。在自动化脚本中加入一些验证点可以实现对预期功能、条件的验证。使用工具使用 `assert(object, value)` 这样的格式向脚本中插入验证点,验证点会被翻译成能被回放框架执行的验证语

句,实现对程序正常条件和异常条件的判断。

[0054] (七)日志

自动化用例往往是以无人执守的方式运行的,其运行结果要通过完整的日志查看。在回放任务完成后,在工具中可以查看回放结果,包括执行用例总数、通过数、失败数、跳过数、失败错误堆栈信息、失败截图等等。同时也可以配置邮箱地址,将日志定时发送给相关负责人。

[0055] (八)、业务流程图,参见图 6。

[0056] 与现有技术相比,本发明的技术方案,至少可以达到以下有益效果:

(1)可以方便的生成自动化脚本,使得测试人员在编写脚本方面的投入大大降低。它还同时兼容 Android 和 iOS 两种主流手机操作系统,使得关注重点放在业务流程上而不是频繁的切换、配置测试环境。

[0057] (2)可以被持续集成调用,在自动编译、做盘后进行黑盒测试,测试结果再持续集成系统读出展现,使开发全流程实现闭环。

[0058] (3)自动化测试可以提高测试效率,使测试人员更加专注于新的测试模块的建立和开发,从而提高测试覆盖率,其次自动化测试更便于测试资产的数字化管理,使得测试资产在整个测试生命周期内可以得到复用,这个特点在功能测试和回归测试中尤其具有意义。此外,测试流程自动化管理可以使机构的测试活动开展更加过程化,符合 CMMI 过程改进的思想。

[0059] 以上结合附图详细说明了本发明的技术方案,考虑到相关技术中没有简便的、统一的针对复杂类型应用测试的解决办法。现有的应用测试无法完成有复杂类型参与的应用测试过程。因此,本发明提出了一种移动应用自动化测试装置和一种移动应用自动化测试方法,可以在现有的应用测试方式基础上,充分利用简单对象类型完成复杂对象类型的应用测试,建立多对象类型参与的应用测试的通用、统一测试思路。

[0060] 以上所述仅为本发明的优选实施例而已,并不用于限制本发明,对于本领域的技术人员来说,本发明可以有各种更改和变化。凡在本发明的精神和原则之内,所作的任何修改、等同替换、改进等,均应包含在本发明的保护范围之内。

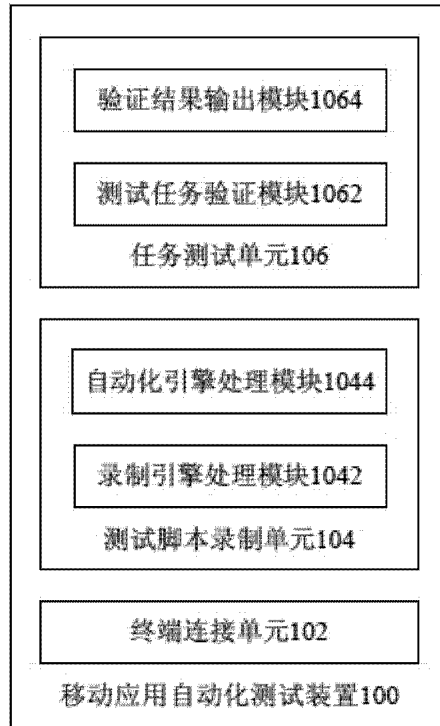


图 1

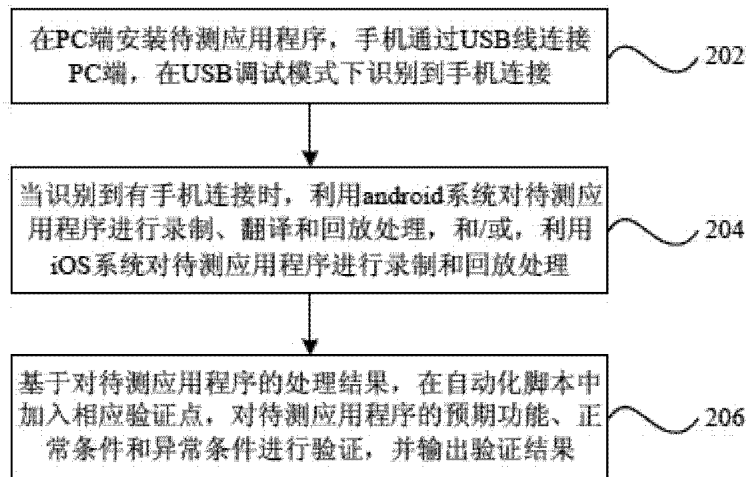


图 2

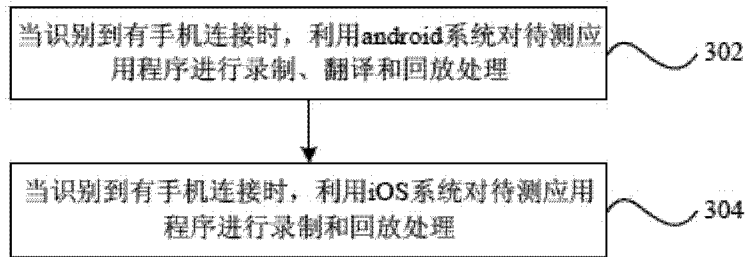


图 3

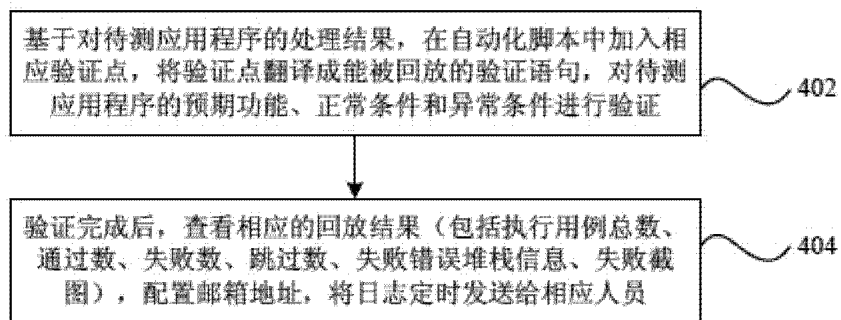


图 4

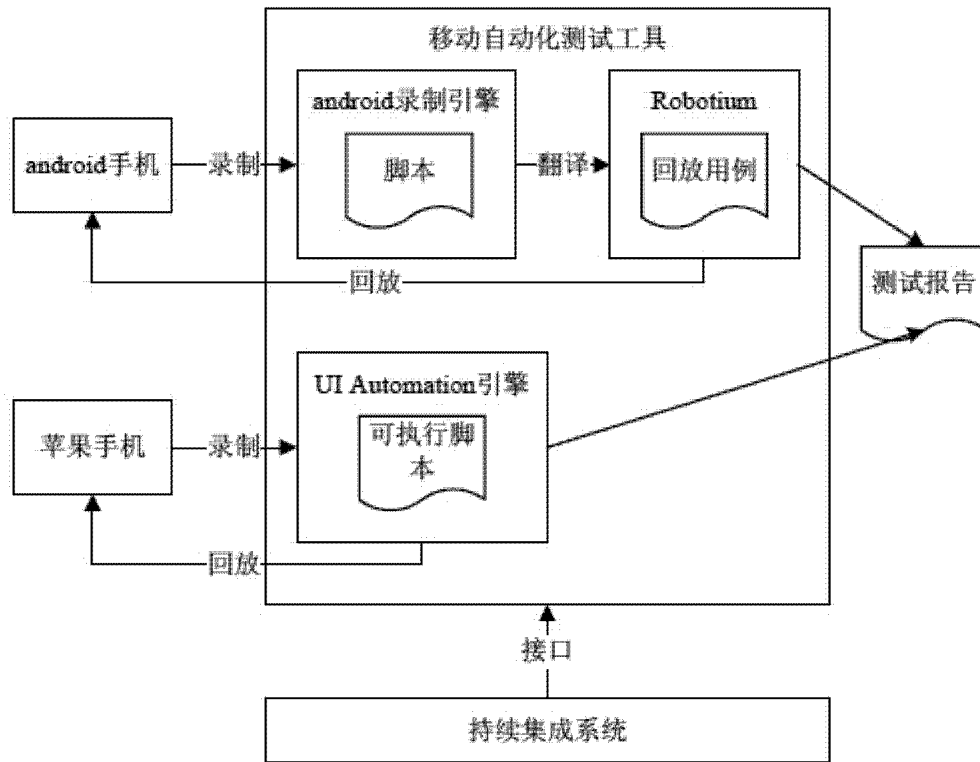


图 5

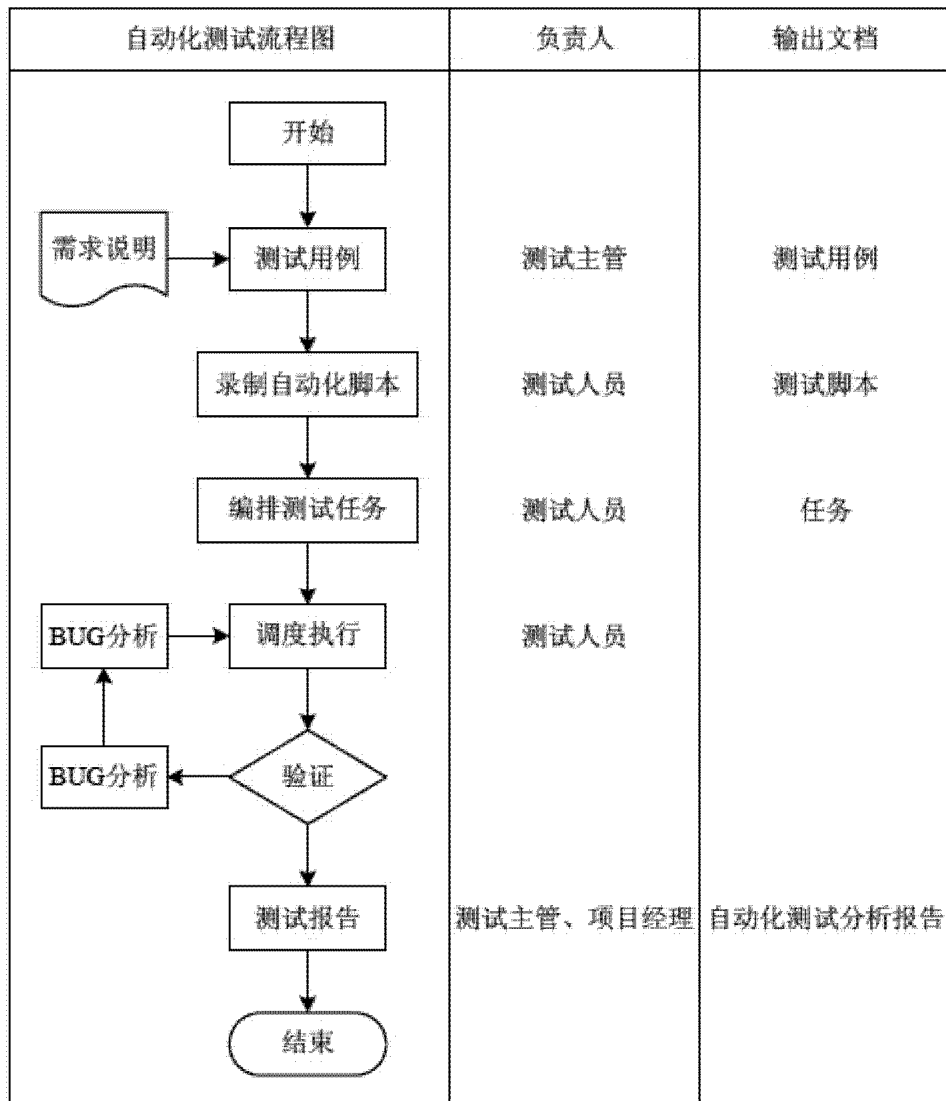


图 6