



(12) 发明专利申请

(10) 申请公布号 CN 112231229 A

(43) 申请公布日 2021.01.15

(21) 申请号 202011239254.0

(22) 申请日 2020.11.09

(71) 申请人 恩亿科(北京)数据科技有限公司  
地址 100192 北京市海淀区西小口路66号  
东升科技园·北领地D-1楼1层136A  
(东升地区)

(72) 发明人 李春

(74) 专利代理机构 青岛清泰联信知识产权代理  
有限公司 37256

代理人 赵燕

(51) Int.Cl.  
G06F 11/36 (2006.01)

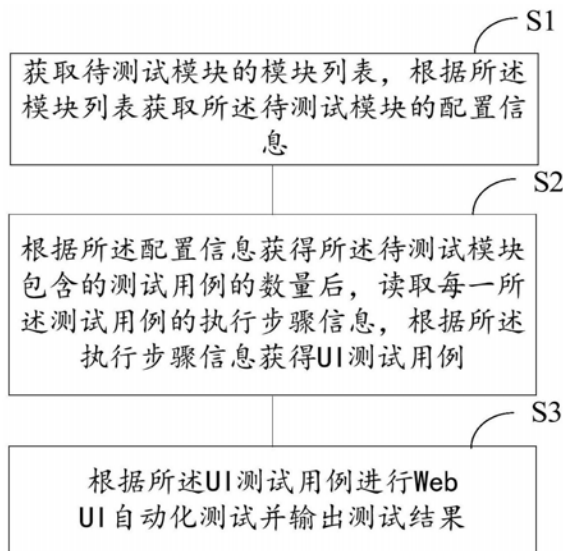
权利要求书2页 说明书8页 附图7页

(54) 发明名称

Web UI自动化测试方法、系统、电子设备及  
可读存储介质

(57) 摘要

本发明公开了一种Web UI自动化测试方法、  
系统、电子设备及可读存储介质,Web UI自动化  
测试方法包括:配置信息获取步骤:获取待测试  
模块的模块列表,根据模块列表获取所述待测试  
模块的配置信息;UI测试用例获得步骤:根据配  
置信息获得待测试模块包含的测试用例的数量  
后,读取每一测试用例的执行步骤信息,根据所  
述执行步骤信息获得UI测试用例;执行步骤:根  
据UI测试用例进行Web UI自动化测试并输出测  
试结果。



1. 一种Web UI自动化测试方法,其特征在于,包括:

配置信息获取步骤:获取待测试模块的模块列表,根据所述模块列表获取所述待测试模块的配置信息;

UI测试用例获得步骤:根据所述配置信息获得所述待测试模块包含的测试用例的数量后,读取每一所述测试用例的执行步骤信息,根据所述执行步骤信息获得UI测试用例;

执行步骤:根据所述UI测试用例进行Web UI自动化测试并输出测试结果。

2. 如权利要求1所述的Web UI自动化测试方法,其特征在于,所述执行步骤信息包含测试用例名称、元素定位方式、元素位置、操作信息以及步骤信息,所述最终测试用例获得步骤包括:

计算步骤:根据所述配置信息获得所述待测试模块包含的测试用例的数量;

读取步骤:读取所述步骤信息及所述步骤信息中的每一步骤对应的所述操作信息;

可能值列表获得步骤:读取并根据所述元素定位方式及所述元素位置定位所述测试用例中的每一元素的位置,获得可能值列表;

UI测试用例输出步骤:根据所述步骤信息及所述操作信息生成正面测试用例及/或负面测试用例,整合所述正面测试用例及/或所述负面测试用例获得所述UI测试用例。

3. 如权利要求2所述的Web UI自动化测试方法,其特征在于,所述最终测试用例输出步骤包括:

判断步骤:对所述正面测试用例及所述负面测试用例的类型进行判断;

正面测试用例处理步骤:当为正面测试用例时,根据正面测试用例名称遍历该步骤下正确的所述步骤信息并将所述步骤信息添加至测试步骤列表;

负面测试用例处理步骤:当为负面测试用例时,判断是否复用正面测试用例,如是则复用正面测试用例重复部分的步骤信息并生成负面测试用例剩余部分的步骤信息,将所述步骤信息添加至测试步骤列表;如不是,则根据负面测试用例名称遍历该步骤下错误的所述步骤信息并将所述步骤信息添加至测试步骤列表;

输出步骤:根据所述测试步骤列表最后组合拼接所述待测试模块的所有所述步骤信息输出所述UI测试用例。

4. 如权利要求2所述的Web UI自动化测试方法,其特征在于,所述执行步骤包括:

匹配步骤:根据所述可能值列表进行元素定位匹配;

正常执行步骤:记录所述最终测试用例的执行状态并生成HTML测试报告;

错误执行步骤:对所述最终测试用例执行错误的地方进行截图,并生成HTML测试报告;

发送步骤:将所述HTML测试报告进行发送。

5. 如权利要求4所述的Web UI自动化测试方法,其特征在于,所述正常执行步骤还包括,输出异常状态下的堆栈轨迹;所述错误执行步骤还包括,输出错误级别的日志。

6. 一种Web UI自动化测试系统,其特征在于,包括:

配置信息获取装置,获取待测试模块的模块列表,根据所述模块列表获取所述待测试模块的配置信息;

UI测试用例获得装置,根据所述配置信息获得所述待测试模块包含的测试用例的数量后,读取每一所述测试用例的执行步骤信息,根据所述执行步骤信息获得UI测试用例;

执行装置,根据所述UI测试用例进行Web UI自动化测试并输出测试结果。

7. 如权利要求6所述的Web UI自动化测试系统,其特征在于,所述执行步骤信息包含测试用例名称、元素定位方式、元素位置、操作信息以及步骤信息,所述最终测试用例获得装置包括:

计算单元,根据所述配置信息获得所述待测试模块包含的测试用例的数量;

读取单元,读取所述步骤信息及所述步骤信息中的每一步骤对应的所述操作信息;

可能值列表获得单元,读取并根据所述元素定位方式及所述元素位置定位所述测试用例中的每一元素的位置,获得可能值列表;

UI测试用例输出单元,根据所述步骤信息及所述操作信息生成正面测试用例及/或负面测试用例,整合所述正面测试用例及/或所述负面测试用例获得所述UI测试用例。

8. 如权利要求7所述的Web UI自动化测试系统,其特征在于,所述最终测试用例输出单元包括:

判断模块:对所述正面测试用例及所述负面测试用例的类型进行判断;

正面测试用例处理模块:当为正面测试用例时,根据正面测试用例名称遍历该步骤下正确的所述步骤信息并将所述步骤信息添加至测试步骤列表;

负面测试用例处理模块:当为负面测试用例时,判断是否复用正面测试用例,如是则复用正面测试用例重复部分的步骤信息并生成负面测试用例剩余部分的步骤信息,将所述步骤信息添加至测试步骤列表;如不是,则根据负面测试用例名称遍历该步骤下错误的所述步骤信息并将所述步骤信息添加至测试步骤列表;

输出模块:根据所述测试步骤列表最后组合拼接所述待测试模块的所有所述步骤信息输出所述最终测试用例。

9. 一种电子设备,包括存储器、处理器以及存储在所述存储器上并可在所述处理器上运行的计算机程序,其特征在于,所述处理器执行所述计算机程序时实现如权利要求1至5中任一项所述的Web UI自动化测试方法。

10. 一种可读存储介质,其上存储有计算机程序,其特征在于,该程序被处理器执行时实现如权利要求1至5中任一项所述的Web UI自动化测试方法。

## Web UI自动化测试方法、系统、电子设备及可读存储介质

### 技术领域

[0001] 本发明涉及UI测试,具体地说,尤其涉及一种Web UI自动化测试方法、系统、电子设备及可读存储介质。

### 背景技术

[0002] 在如今系统设计的背景下,WebUI的交互越来越复杂,针对该类型的UI测试需求也日趋旺盛。

[0003] 在传统的UI测试模式下,测试人员需要基本使用手工测试,在界面上进行操作,并且测试人员还需要维护大量的测试用例与测试数据。对于成型的系统来说,主体业务逻辑的UI测试存在一定的逻辑规律,并且针对同类型的栏位配置信息,测试用例的相似度非常高,变动的可能是元素的位置。

[0004] 为了应对频繁的系统新需求或变更需求,测试人员往往需要花费大量人力在回归测试用例上,其中很多都是单一重复性的操作。尤其是当某个模块的界面做了调整或者发生改变时,往往回归调试对应的测试用例需要占用大量的时间,而紧凑的开发进度可能会导致这部分相关联的模块无法得到充分的测试,从而在生产环境上线后发现新的缺陷。

[0005] 因此急需开发一种克服上述缺陷的Web UI自动化测试方法、系统、电子设备及可读存储介质。

### 发明内容

[0006] 针对上述问题,本发明提供一种Web UI自动化测试方法,其中,包括:

[0007] 配置信息获取步骤:获取待测试模块的模块列表,根据所述模块列表获取所述待测试模块的配置信息;

[0008] UI测试用例获得步骤:根据所述配置信息获得所述待测试模块包含的测试用例的数量后,读取每一所述测试用例的执行步骤信息,根据所述执行步骤信息获得UI测试用例;

[0009] 执行步骤:根据所述UI测试用例进行Web UI自动化测试并输出测试结果。

[0010] 上述的Web UI自动化测试系统,其中,所述执行步骤信息包含测试用例名称、元素定位方式、元素位置、操作信息以及步骤信息,所述最终测试用例获得步骤包括:

[0011] 计算步骤:根据所述配置信息获得所述待测试模块包含的测试用例的数量;

[0012] 读取步骤:读取所述步骤信息及所述步骤信息中的每一步骤对应的所述操作信息;

[0013] 可能值列表获得步骤:读取并根据所述元素定位方式及所述元素位置定位所述测试用例中的每一元素的位置,获得可能值列表;

[0014] 上述的Web UI自动化测试方法,其中,所述最终测试用例输出步骤包括:

[0015] 判断步骤:对所述正面测试用例及所述负面测试用例的类型进行判断;

[0016] 正面测试用例处理步骤:当为正面测试用例时,根据正面测试用例名称遍历该步骤下正确的所述步骤信息并将所述步骤信息添加至测试步骤列表;

[0017] 负面测试用例处理步骤:当为负面测试用例时,判断是否复用正面测试用例,如是则复用正面测试用例重复部分的步骤信息并生成负面测试用例剩余部分的步骤信息,将所述步骤信息添加至测试步骤列表;如不是,则根据负面测试用例名称遍历该步骤下错误的所述步骤信息并将所述步骤信息添加至测试步骤列表;

[0018] 输出步骤:根据所述测试步骤列表最后组合拼接所述待测试模块的所有所述步骤信息输出所述最终测试用例。

[0019] 上述的Web UI自动化测试方法,其中,所述执行步骤包括:

[0020] 匹配步骤:根据所述可能值列表进行元素定位匹配;

[0021] 正常执行步骤:记录所述最终测试用例的执行状态并生成HTML测试报告;

[0022] 错误执行步骤:对所述最终测试用例执行错误的地方进行截图,并生成HTML测试报告;

[0023] 发送步骤:将所述HTML测试报告进行发送。

[0024] 上述的Web UI自动化测试方法,其中,所述正常执行步骤还包括,输出异常状态下的堆栈轨迹;所述错误执行步骤还包括,输出错误级别的日志。

[0025] 本发明还提供一种Web UI自动化测试系统,其中,包括:

[0026] 配置信息获取装置,获取待测试模块的模块列表,根据所述模块列表获取所述待测试模块的配置信息;

[0027] UI测试用例获得装置,根据所述配置信息获得所述待测试模块包含的测试用例的数量后,读取每一所述测试用例的执行步骤信息,根据所述执行步骤信息获得UI测试用例;

[0028] 执行装置,根据所述UI测试用例进行Web UI自动化测试并输出测试结果。

[0029] 上述的Web UI自动化测试系统,其中,所述执行步骤信息包含测试用例名称、元素定位方式、元素位置、操作信息以及步骤信息,所述最终测试用例获得装置包括:

[0030] 计算单元,根据所述配置信息获得所述待测试模块包含的测试用例的数量;

[0031] 读取单元,读取所述步骤信息及所述步骤信息中的每一步骤对应的所述操作信息;

[0032] 可能值列表获得单元,读取并根据所述元素定位方式及所述元素位置定位所述测试用例中的每一元素的位置,获得可能值列表;

[0033] UI测试用例输出单元,根据所述步骤信息及所述操作信息生成正面测试用例及/或负面测试用例,整合所述正面测试用例及/或所述负面测试用例获得所述UI测试用例。

[0034] 上述的Web UI自动化测试系统,其中,所述最终测试用例输出单元包括:

[0035] 判断模块:对所述正面测试用例及所述负面测试用例的类型进行判断;

[0036] 正面测试用例处理模块:当为正面测试用例时,根据正面测试用例名称遍历该步骤下正确的所述步骤信息并将所述步骤信息添加至测试步骤列表;

[0037] 负面测试用例处理模块:当为负面测试用例时,判断是否复用正面测试用例,如是则复用正面测试用例重复部分的步骤信息并生成负面测试用例剩余部分的步骤信息,将所述步骤信息添加至测试步骤列表;如不是,则根据负面测试用例名称遍历该步骤下错误的所述步骤信息并将所述步骤信息添加至测试步骤列表;

[0038] 输出模块:根据所述测试步骤列表最后组合拼接所述待测试模块的所有所述步骤信息输出所述最终测试用例。

[0039] 本发明还提供一种电子设备,包括存储器、处理器以及存储在所述存储器上并可在所述处理器上运行的计算机程序,其中,所述处理器执行所述计算机程序时实现如上述中任一项所述的Web UI自动化测试方法。

[0040] 本发明还提供一种可读存储介质,其上存储有计算机程序,其中,该程序被处理器执行时实现如上述中任一项所述的Web UI自动化测试方法。

[0041] 综上所述,本发明相对于现有技术其功效在于:本发明可自动读取测试用例每一步骤,动态生成UI测试用例,从而能够解决面对频繁迭代的系统需求,UI测试需要大量回归的问题。同时本发明可减少测试人员的重复操作,节省时间。并且本发明可管理并自动维护UI测试用例,帮助不会编写代码的测试人员进行WebUI自动化测试。

[0042] 本发明的其它特征和优点将在随后的说明书中阐述,并且,部分地从说明书中变得显而易见,或者通过实施本发明而了解。本发明的目的和其他优点可通过在说明书、权利要求书以及附图中所指出的结构来实现和获得。

## 附图说明

[0043] 为了更清楚地说明本发明实施例或现有技术中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图作一简单地介绍,显而易见地,下面描述中的附图是本发明的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

[0044] 图1为本发明的Web UI自动化测试方法的流程图;

[0045] 图2为图1中步骤S2的分步骤流程图;

[0046] 图3为图2中步骤S24的分步骤流程图;

[0047] 图4为元素定位流程图;

[0048] 图5为步骤S24的应用流程图;

[0049] 图6为图1中步骤S3的分步骤流程图;

[0050] 图7为图6中步骤S32、S33及S34的应用流程图;

[0051] 图8为图6中步骤S31的应用流程图;

[0052] 图9为本发明的Web UI自动化测试系统的结构示意图;

[0053] 图10为本发明的电子设备的结构示意图。

## 具体实施方式

[0054] 为使本发明实施例的目的、技术方案和优点更加清楚,下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行清楚、完整地说明,显然,所描述的实施例是本发明一部分实施例,而不是全部的实施例。基于本发明中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都属于本发明保护的范围。

[0055] 本发明的示意性实施例及其说明用于解释本发明,但并不作为对本发明的限定。另外,在附图及实施方式中所使用相同或类似标号的元件/构件是用来代表相同或类似部分。

[0056] 关于本文中所使用的“第一”、“第二”、“S1”、“S2”、…等,并非特别指称次序或顺位的意思,也非用以限定本发明,其仅为了区别以相同技术用语描述的元件或操作。

[0057] 关于本文中所使用的方向用语,例如:上、下、左、右、前或后等,仅是参考附图的方向。因此,使用的方向用语是用来说明并非用来限制本创作。

[0058] 关于本文中所使用的“包含”、“包括”、“具有”、“含有”等等,均为开放性的用语,即意指包含但不限于。

[0059] 关于本文中所使用的“及/或”,包括所述事物的任一或全部组合。

[0060] 关于本文中的“多个”包括“两个”及“两个以上”;关于本文中的“多组”包括“两组”及“两组以上”。

[0061] 关于本文中所使用的用语“大致”、“约”等,用以修饰任何可以微变化的数量或误差,但这些微变化或误差并不会改变其本质。一般而言,此类用语所修饰的微变化或误差的范围在部分实施例中可为20%,在部分实施例中可为10%,在部分实施例中可为5%或是其他数值。本领域技术人员应当了解,前述提及的数值可依实际需求而调整,并不以此为限。

[0062] 某些用以描述本申请的用词将于下或在此说明书的别处讨论,以提供本领域技术人员在有关本申请的描述上额外的引导。

[0063] 本发明针对WebUI的自动化测试,在模拟人工操作的基础之上,基于人工操作的步骤,维护形成测试用例,使用Python语言并结合Selenium框架,读取测试用例,按维护的测试用例来自动执行每一步的操作,减少重复的劳动,减轻测试人员的日常工作负担,用以节省不必要的维护成本。其中,Web UI自动化测试的输入项为实现维护好的测试用例,读取每一步的操作,根据不同模块的操作细化成每一步骤,生成可直接执行的测试用例,覆盖超过90%的Web UI自动化测试场景。在配置信息发生变动时,可读取变动信息,自动执行,只需手动更新维护测试用例就可。以下结合附图进行具体说明。

[0064] 请参照图1,图1为本发明的Web UI自动化测试方法的流程图。如图1所示,本发明的Web UI自动化测试方法,包括:

[0065] 配置信息获取步骤S1:获取待测试模块的模块列表,根据所述模块列表获取所述待测试模块的配置信息;

[0066] UI测试用例获得步骤S2:根据所述配置信息获得所述待测试模块包含的测试用例的数量后,读取每一所述测试用例的执行步骤信息,根据所述执行步骤信息获得UI测试用例;

[0067] 执行步骤S3:根据所述UI测试用例进行Web UI自动化测试并输出测试结果。

[0068] 其中,所述执行步骤信息包含测试用例名称、元素定位方式、元素位置、操作信息、步骤信息以及输入参数信息,具体地说,本发明使用Python语言,读取Excel维护的UI执行步骤信息,包含测试用例名称,元素定位方式,元素位置、元素操作以及输入参数信息。本发明根据不同的元素位置以及元素操作,通过测试用例名称逻辑分组,生成一系列UI测试用例。由此,本发明可自动读取测试用例每一步骤,动态生成UI测试用例,具体地说,针对待测试的模块列表,本发明逐个读取各个待测模块的配置信息,计算出每个待测模块包含的测试用例数量,根据每个测试用例配置的执行步骤信息,生成测试用例并执行。从而解决面对频繁迭代的系统需求,UI测试需要大量回归的问题;同时可减少测试人员的重复操作,节省时间,还可管理并自动维护UI测试用例,帮助不会编写代码的测试人员进行WebUI自动化测试。

[0069] 请参照图2,图2为图1中步骤S2的分步骤流程图。如图2所示,所述最终测试用例获

得步骤S2包括:

[0070] 计算步骤S21:根据所述配置信息获得所述待测试模块包含的测试用例的数量。

[0071] 读取步骤S22:读取所述步骤信息及所述步骤信息中的每一步骤对应的所述操作信息。

[0072] 可能值列表获得步骤S23:读取并根据所述元素定位方式及所述元素位置定位所述测试用例中的每一元素的位置,获得可能值列表。

[0073] UI测试用例输出步骤S24:根据所述步骤信息及所述操作信息生成正面测试用例及/或负面测试用例,整合所述正面测试用例及/或所述负面测试用例获得所述UI测试用例。

[0074] 其中,请参照图4,图4为元素定位流程图,如图4所示,在可能值列表获得步骤S23中,以正面测试用例为例,在生成用例时会依次判断元素是否具有唯一特征,相对位置是否固定,例如,通过ID、xpath、selector、classname、linktext、partial link text及tagname生成元素定位方式,如果界面上新增了元素,该元素是否仍能定位到等信息,依次生成该元素所有的可能的定位方式范围,并放入可能值列表,供后续使用,其中负面测试用例的逻辑与正面测试用例的逻辑相同,因此在此就不再赘述了。

[0075] 请参照图3及图5,图3为图2中步骤S24的分步骤流程图,图5为步骤S24的应用流程图。如图3及图5所示,所述最终测试用例输出步骤S24包括:

[0076] 判断步骤S241:对所述正面测试用例及所述负面测试用例的类型进行判断;

[0077] 正面测试用例处理步骤S242:当为正面测试用例时,根据正面测试用例名称遍历该步骤下正确的所述步骤信息并将所述步骤信息添加至测试步骤列表;

[0078] 负面测试用例处理步骤S243:当为负面测试用例时,判断是否复用正面测试用例,如是则复用正面测试用例重复部分的步骤信息并生成负面测试用例剩余部分的步骤信息,将所述步骤信息添加至测试步骤列表;如不是,则根据负面测试用例名称遍历该步骤下错误的所述步骤信息并将所述步骤信息添加至测试步骤列表;

[0079] 其中,在本实施例中,生成测试用例时,会生成正面用例以及负面用例。正面用例生成时,获取正面用例可用的测试步骤列表,逐个放入测试用例中。负面用例生成时,获取负面用例可用的测试步骤列表,逐个放入测试用例中。最后组合拼接模块的所有测试用例信息,生成一个完整的测试用例列表。

[0080] 输出步骤S244:根据所述测试步骤列表最后组合拼接所述待测试模块的所有所述步骤信息输出所述UI测试用例。

[0081] 请参照图6及图7,图6为图1中步骤S3的分步骤流程图,图7为图6中步骤S32、S33及S34的应用流程图。如图6及图7所示,所述执行步骤S3包括:

[0082] 匹配步骤S31:根据所述可能值列表进行元素定位匹配;

[0083] 正常执行步骤S32:记录所述最终测试用例的执行状态并生成HTML测试报告;

[0084] 错误执行步骤S33:对所述最终测试用例执行错误的地方进行截图,并生成HTML测试报告;

[0085] 发送步骤S34:将所述HTML测试报告进行发送。

[0086] 其中,所述正常执行步骤还包括,输出异常状态下的堆栈轨迹;所述错误执行步骤还包括,输出错误级别的日志。



[0087] WebUI自动化测试过程中,经常会遇到元素定位不到情况,通常报错信息为“no such element”,通常引起该情况可以从以下几方面入手,一是网络是否存在延迟,整个页面没有被完全加载。二是遇到iframe的结构,需要切换到iframe。三是元素在不同的窗口,需要判断当前条件下有几个窗口,然后在切换句柄。四是页面返回了动态id,通过浏览器自动的定位方式往往会定位不到元素,此时可以通过fullxpath来代替。请参照图8,图8为图6中步骤S31的应用流程图。如图8所示,在匹配步骤S31中,首先匹配可能值列表,如果能定位到元素,则进行下一步处理;如果定位不到元素,依次通过添加智能等待、切换到对应的iframe、轮询句柄切换到对应的窗口及xfullpath这些方式对元素进行定位,定位成功则进行下一步。

[0088] 请参照图9,图9为本发明的Web UI自动化测试系统的结构示意图。如图9所示,本发明的Web UI自动化测试系统,包括:

[0089] 配置信息获取装置11,获取待测试模块的模块列表,根据所述模块列表获取所述待测试模块的配置信息;

[0090] UI测试用例获得装置12,根据所述配置信息获得所述待测试模块包含的测试用例的数量后,读取每一所述测试用例的执行步骤信息,根据所述执行步骤信息获得UI测试用例;

[0091] 执行装置13,根据所述UI测试用例进行Web UI自动化测试并输出测试结果。

[0092] 其中,所述执行步骤信息包含测试用例名称、元素定位方式、元素位置、操作信息以及步骤信息。

[0093] 进一步地,所述最终测试用例获得装置12包括:

[0094] 计算单元121,根据所述配置信息获得所述待测试模块包含的测试用例的数量;

[0095] 读取单元122,读取所述步骤信息及所述步骤信息中的每一步骤对应的所述操作信息;

[0096] 可能值列表获得单元123,读取并根据所述元素定位方式及所述元素位置定位所述测试用例中的每一元素的位置,获得可能值列表;

[0097] UI测试用例输出单元124,根据所述步骤信息及所述操作信息生成正面测试用例及/或负面测试用例,整合所述正面测试用例及/或所述负面测试用例获得所述UI测试用例。

[0098] 再进一步地,所述最终测试用例输出单元124包括:

[0099] 判断模块1241:对所述正面测试用例及所述负面测试用例的类型进行判断;

[0100] 正面测试用例处理模块1242:当为正面测试用例时,根据正面测试用例名称遍历该步骤下正确的所述步骤信息并将所述步骤信息添加至测试步骤列表;

[0101] 负面测试用例处理模块1243:当为负面测试用例时,判断是否复用正面测试用例,如是则复用正面测试用例重复部分的步骤信息并生成负面测试用例剩余部分的步骤信息,将所述步骤信息添加至测试步骤列表;如不是,则根据负面测试用例名称遍历该步骤下错误的所述步骤信息并将所述步骤信息添加至测试步骤列表;

[0102] 输出模块1244:根据所述测试步骤列表最后组合拼接所述待测试模块的所有所述步骤信息输出所述最终测试用例。

[0103] 请参照图10,图10为本发明的电子设备的结构示意图。如图10所示,

[0104] 本实施例揭示了一种电子设备的一种具体实施方式。电子设备可以包括处理器81以及存储有计算机程序指令的存储器82。

[0105] 具体地,上述处理器81可以包括中央处理器(CPU),或者特定集成电路(Application Specific Integrated Circuit,简称为ASIC),或者可以被配置成实施本申请实施例的一个或多个集成电路。

[0106] 其中,存储器82可以包括用于数据或指令的大容量存储器。举例来说而非限制,存储器82可包括硬盘驱动器(Hard Disk Drive,简称为HDD)、软盘驱动器、固态驱动器(Solid State Drive,简称为SSD)、闪存、光盘、磁光盘、磁带或通用串行总线(Universal Serial Bus,简称为USB)驱动器或者两个或更多个以上这些的组合。在合适的情况下,存储器82可包括可移除或不可移除(或固定)的介质。在合适的情况下,存储器82可在数据处理装置的内部或外部。在特定实施例中,存储器82是非易失性(Non-Volatile)存储器。在特定实施例中,存储器82包括只读存储器(Read-Only Memory,简称为ROM)和随机存取存储器(Random Access Memory,简称为RAM)。在合适的情况下,该ROM可以是掩模编程的ROM、可编程ROM(Programmable Read-Only Memory,简称为PROM)、可擦除PROM(Erasable Programmable Read-Only Memory,简称为EPROM)、电可擦除PROM(Electrically Erasable Programmable Read-Only Memory,简称为EEPROM)、电可改写ROM(Electrically Alterable Read-Only Memory,简称为EAROM)或闪存(FLASH)或者两个或更多个以上这些的组合。在合适的情况下,该RAM可以是静态随机存取存储器(Static Random-Access Memory,简称为SRAM)或动态随机存取存储器(Dynamic Random Access Memory,简称为DRAM),其中,DRAM可以是快速页模式动态随机存取存储器(Fast Page Mode Dynamic Random Access Memory,简称为FPMDRAM)、扩展数据输出动态随机存取存储器(Extended Date Out Dynamic Random Access Memory,简称为EDODRAM)、同步动态随机存取内存(Synchronous Dynamic Random-Access Memory,简称SDRAM)等。

[0107] 存储器82可以用来存储或者缓存需要处理和/或通信使用的各种数据文件,以及处理器81所执行的可能的计算机程序指令。

[0108] 处理器81通过读取并执行存储器82中存储的计算机程序指令,以实现上述实施例中的任意一种文件系统容量管理优化方法、。

[0109] 在其中一些实施例中,电子设备还可包括通信接口83和总线80。其中,如图10所示,处理器81、存储器82、通信接口83通过总线80连接并完成相互间的通信。

[0110] 通信接口83用于实现本申请实施例中各模块、装置、单元和/或设备之间的通信。通信端口83还可以实现与其他部件例如:外接设备、图像/数据采集设备、数据库、外部存储以及图像/数据处理工作站等之间进行数据通信。

[0111] 总线80包括硬件、软件或两者,将电子设备的部件彼此耦接在一起。总线80包括但不限于以下至少之一:数据总线(Data Bus)、地址总线(Address Bus)、控制总线(Control Bus)、扩展总线(Expansion Bus)、局部总线(Local Bus)。举例来说而非限制,总线80可包括图形加速接口(Accelerated Graphics Port,简称为AGP)或其他图形总线、增强工业标准架构(Extended Industry Standard Architecture,简称为EISA)总线、前端总线(Front Side Bus,简称为FSB)、超传输(Hyper Transport,简称为HT)互连、工业标准架构(Industry Standard Architecture,简称为ISA)总线、无线带宽(InfiniBand)互连、低引

脚数 (Low Pin Count, 简称为LPC) 总线、存储器总线、微信道架构 (Micro Channel Architecture, 简称为MCA) 总线、外围组件互连 (Peripheral Component Interconnect, 简称为PCI) 总线、PCI-Express (PCI-X) 总线、串行高级技术附件 (Serial Advanced Technology Attachment, 简称为SATA) 总线、视频电子标准协会局部 (Video Electronics Standards Association Local Bus, 简称为VLB) 总线或其他合适的总线或者两个或更多个以上这些的组合。在合适的情况下, 总线80可包括一个或多个总线。尽管本申请实施例描述和示出了特定的总线, 但本申请考虑任何合适的总线或互连。

[0112] 另外, 结合上述实施例中WebUI自动化测试方法, 本申请实施例可提供一种计算机可读存储介质来实现。该计算机可读存储介质上存储有计算机程序指令; 该计算机程序指令被处理器执行时实现上述实施例中的任意一种WebUI自动化测试方法。

[0113] 综上所述, 本发明具有以下效果:

[0114] 1. 测试人员无需再手动去操作UI执行回归测试, 本技术可根据文本配置UI操作信息, 一键生成所有的测试用例, 覆盖超过90%的UI测试场景, 减少测试人员的重复操作, 节省时间;

[0115] 2. 在UI元素位置发生变化时, 测试人员只需手动更新测试元素位置信息。测试人员可通过本技术读取测试用例的变化信息, 根据最新的配置信息执行测试用例;

[0116] 3. 不会编写代码的测试人员也可通过本技术, 简单实现Web UI自动化测试。

[0117] 尽管参照前述实施例对本发明进行了详细的说明, 本领域的普通技术人员应当理解: 其依然可以对前述各实施例所记载的技术方案进行修改, 或者对其中部分技术特征进行等同替换; 而这些修改或者替换, 并不使相应技术方案的本质脱离本发明各实施例技术方案的精神和范围。

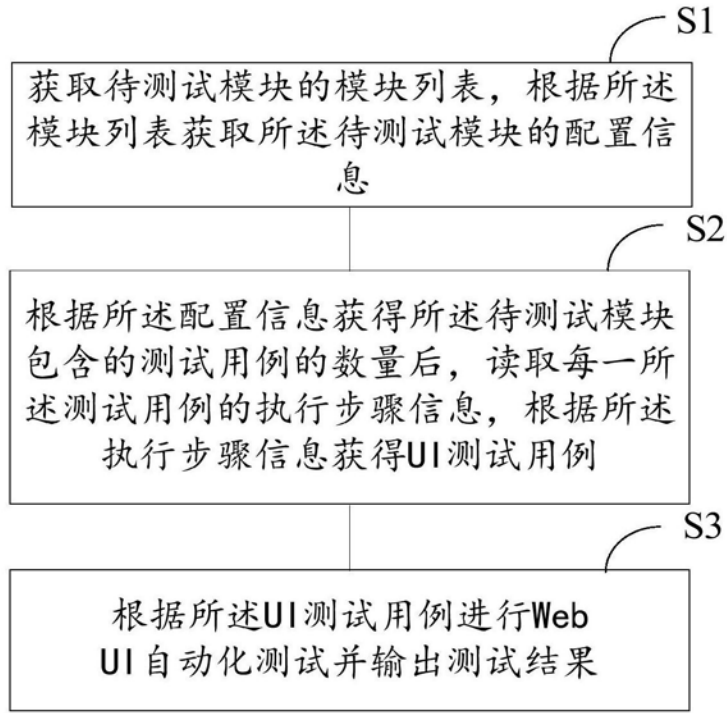


图1

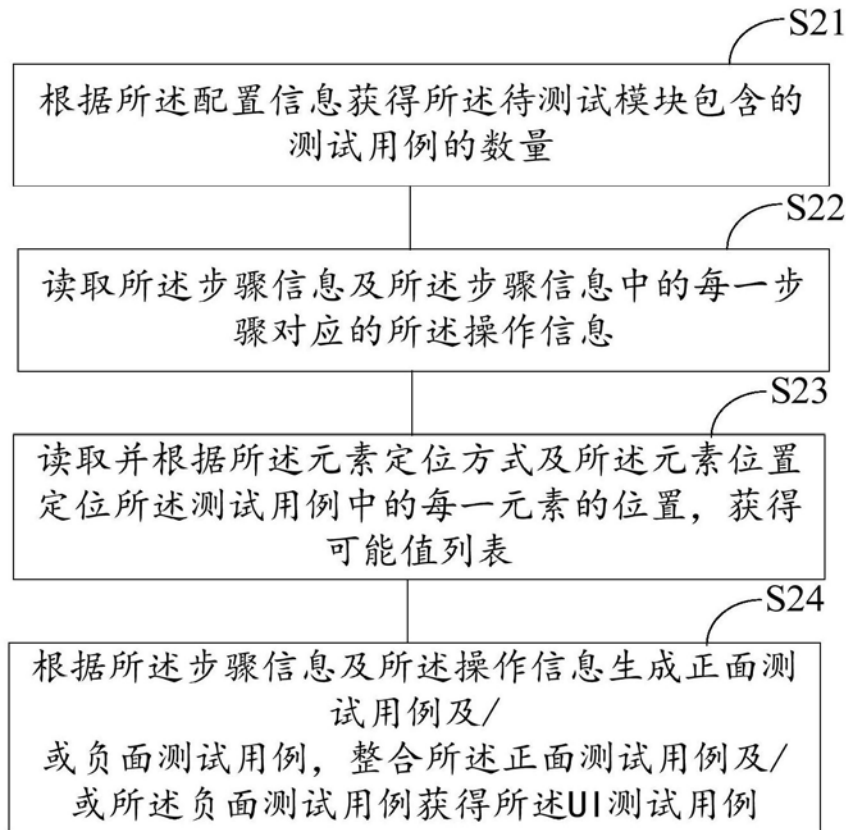


图2

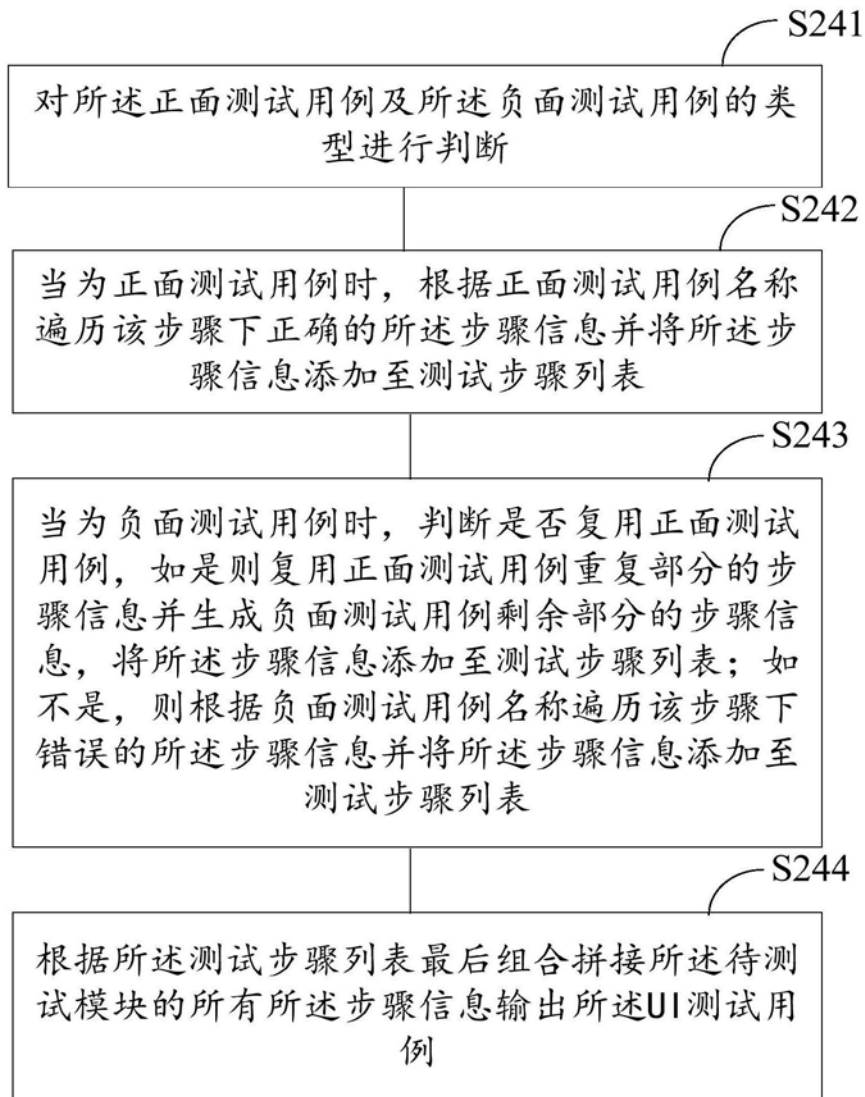


图3

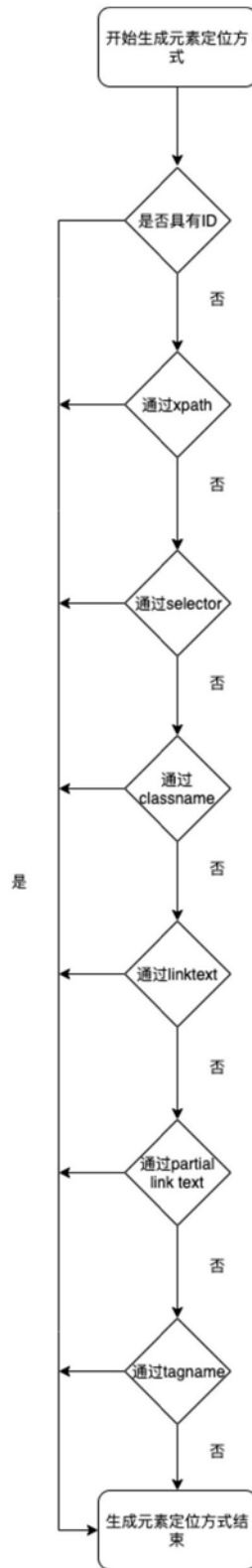


图4

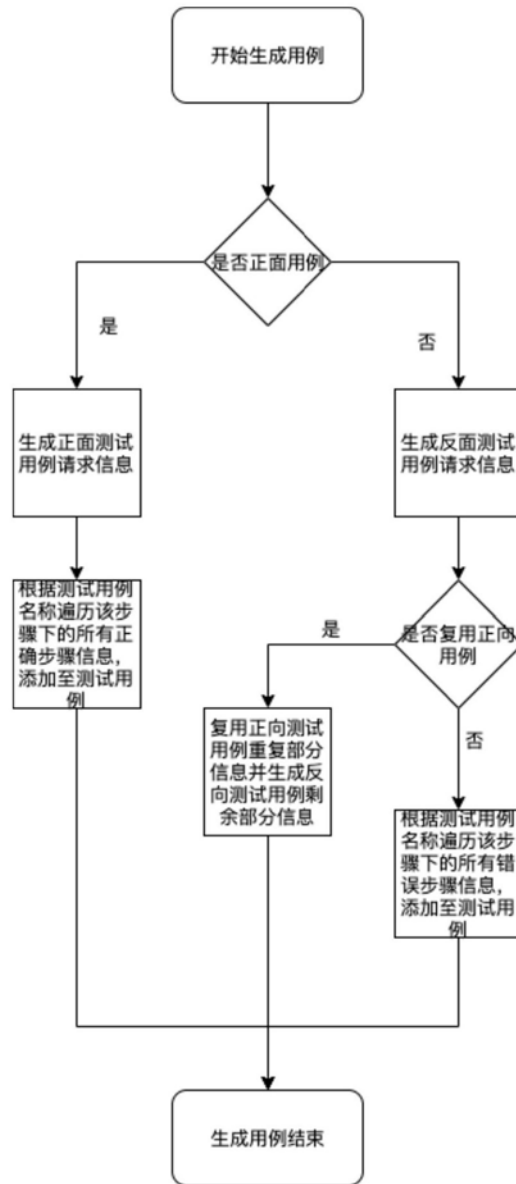


图5

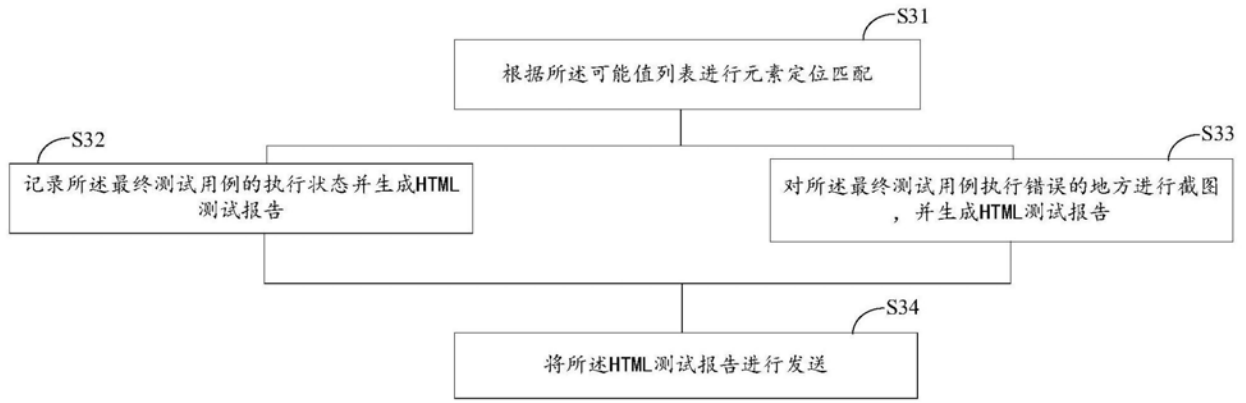


图6

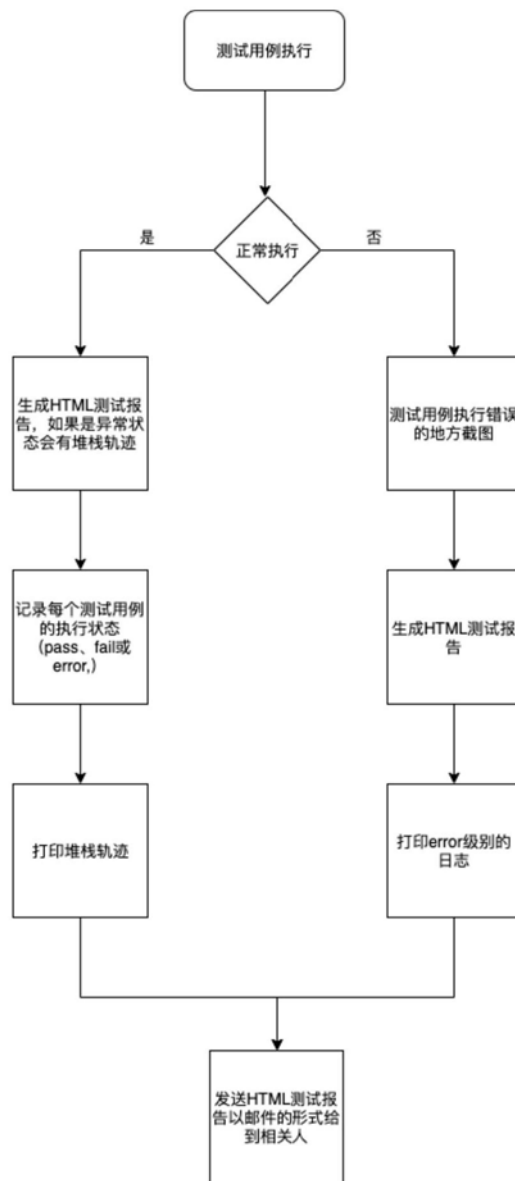


图7



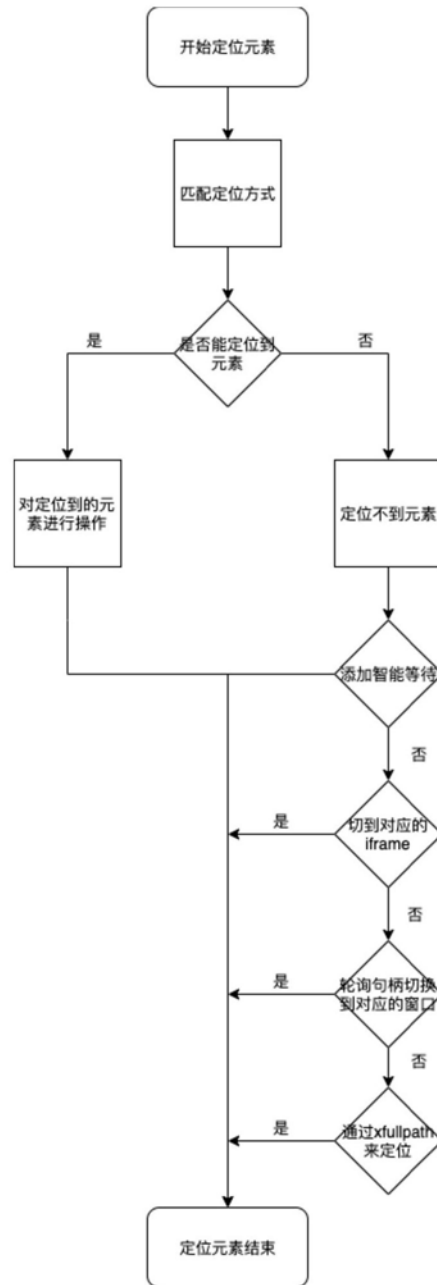


图8

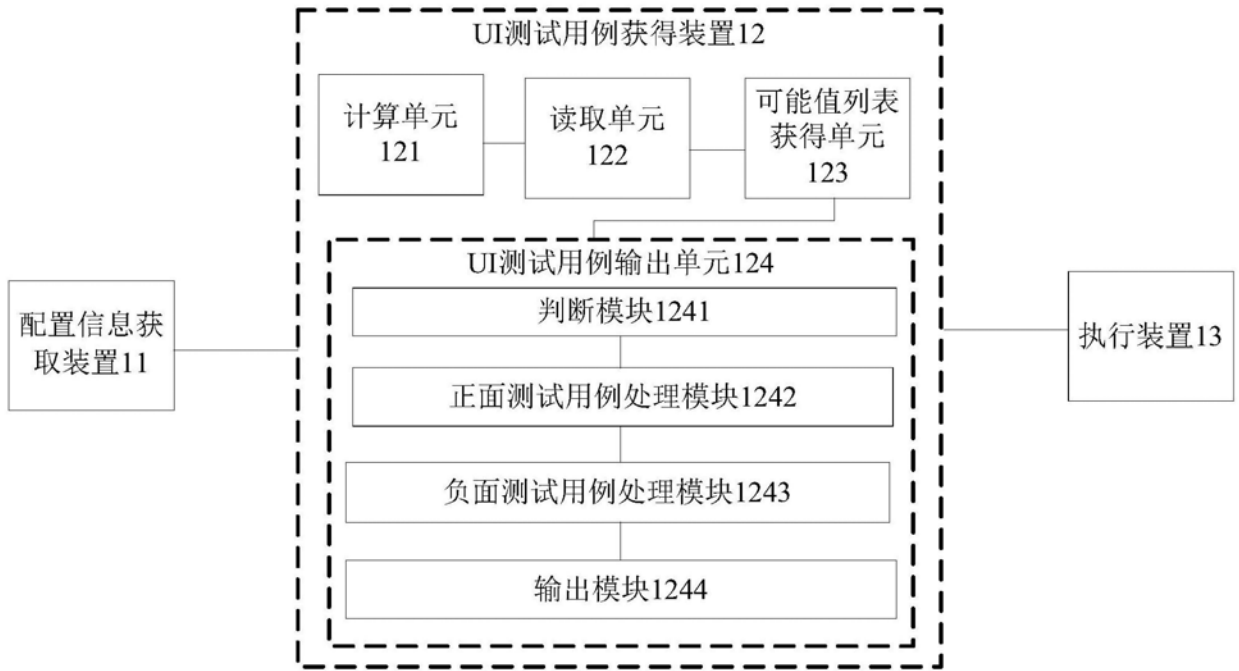


图9

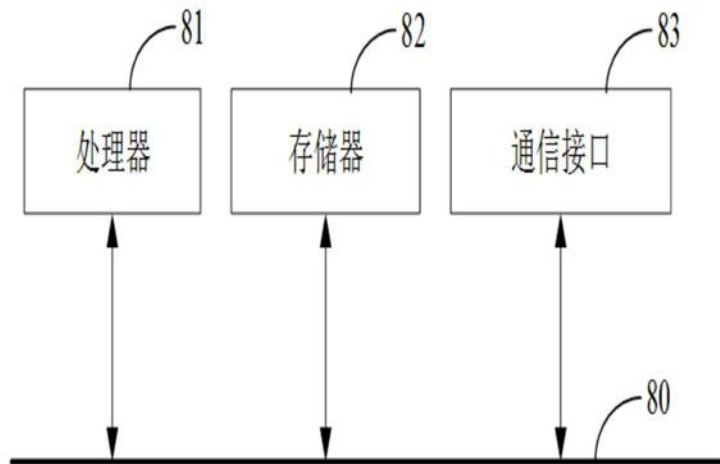


图10