Inline Text Wrapping Picture

北京邮电大学

硕士研究生学位论文开题报告

学 号: 2017140655

姓 名: 陈胜强

学 院: 软件学院

专业(领域): 软件工程

研究方向:

导师姓名: 崔毅东

攻 读 学 位: 工程硕士

2018年11月21日

|  |  |  |  |
| --- | --- | --- | --- |
| 论文题目 | 移动应用的UI自动化测试系统的设计与实现 | | |
| 选题来源 | 973、863项目 | 论文类型 | 基础研究 |
| 开题日期 | 2018-11-20 | 开题地点 | 北京邮电大学 |
| **一、立题依据（包括研究目的、意义、国内外研究现状和发展趋势，需结合科学研究发展趋势来论述科学意义；或结合国民经济和社会发展中迫切需要解决的关键科技问题来论述其应用前景。附主要参考文献目录）（不少于800字）**  1.研究背景  移动设备发展到今天，极大方便了人们的生活，无论是工作学习还是娱乐休闲，都需要各式各样的app来满足日常生活。于是，app如雨后春笋浮现在公众的视野，但是app数量以及迭代版本的迅速飙升给人生活带来便利的同时，对于app的开发以及上线前的测试工作也带来不可忽视的影响。如果对于app的测试工作还局限于人工测试的话势必会影响app的上线效率，因此必须让机器去承担绝大多数的测试任务来解放人力，提高效率，自动化遍历应运而生。  2.国内外研究现状和发展趋势  就目前来看，大环境中已经有许多自动化测试的工具/平台，各大公司也有属于自己的一套商业模式。比如百度MTC-移动云测试中心提出了以下口号：提供简单易用，快速高效的测试服务，为开发者解决测试过程中的设备，人力，技术等问题，助力移动开发者提升产品质量和用户体验；腾讯WeTest-质量开放平台也提出：提供云端自动化兼容服务，提交云端百台真机，并行测试。快速发现游戏/应用兼容性和性能问题，覆盖安卓主流机型；国外的EvoSuite（Automatic Test Suite Generation for Java）目前也仅仅针对java程序生成单元测试，并不能进行其它测试；test.ai已经开始尝试在测试领域加入人工智能来提升测试效率。显然这些公司并不能覆盖行业大多数用户的要求，提供的客户的也仅限于基于monkey的测试，对于UI的遍历算法也不完备，同时也无法将完整的测试用例反馈给用户，用户也就无法复现一个完整的测试过程，对于用户来说，是不友好的。  总的来看，国内外在测试领域都在加大人力物力的投资，拼命抢夺筹码，争取在激烈的市场竞争中占领些许份额。随着移动设备的普及，未来行业对于自动化测试的依赖会越来越大，一个好的测试工具会引领测试的未来方向。  3.研究意义  当下，功能已经完整和成熟的app，迭代新版本的频率很快，需求稳定，不会频繁变更，因此新老版本之间的变化并不是很大，在这种情况下，自动化测试相较于人工测试则效率显得更加高效；而对于刚刚开发出来的app，自动化测试则能用比较短的时间完成大部分人工测试的任务，一套完整的测试流程也会事半功倍，满足用户需求的同时，又提高了效率。  自动化用执行次数来增加价值，执行次数越多，自动化价值越大。  特别是那些需要重复进行UI操作，比如适配测试，需要适配几十个机型，显然用自动化去执行要优于一个个的手工执行。  UI自动化主要作用是保证业务流程的贯通  UI自动化能够帮助我们确保不会出现一些死机的问题，比如登录不成功，页面打不开等等。  4.参考文献  [1]温素剑，零成本实现Android/iOS自动化测试 基于Appium和Test Perfect [M]，电子工业出版社，2016-01-01  [2][美] Donald G. Firesmith 著，王文慧 译，测试反模式：有效规避常见的92种测试陷阱 [M]，机械工业出版社，2015-01-01  [3][美] 梅耶（Myers, G. J.）等著, 张晓明，黄琳 译 软件测试的艺术 [M]，机械工业出版社  [4][美] WilliamE.Perry，软件测试的有效方法 [M]，清华大学出版社  [5]Tim Riley / Adam Goucher著，张奭、鲍臣礼[1] 译 测试之美[M]，机械工程出版社  [6][美] 培智 著，鲍臣礼等[1] 译 微软的软件测试之道 [M]，机械工业出版社  [7]V.S.Sundara Rajan,A.Malini,K.Sundarakantham,Performance evaluation of online mobile application using Test My App [J],ICACCCT,2014  [8]AZIM T, NEAMTIU I,Targeted and depth-first exploration for systematic testing of android apps[J],ACM SIGPLAN Notices,2013,48(10):641-660  [9]Appium[EB/OL],[2016-01-20],https://github.com/Appium/Appium  [10]M.L. Vásquez,K. Moran,D. Poshyvanyk, Continuous, Evolutionary and Large-Scale: A New Perspective for Automated Mobile App Testing [J],ICSME,2017,27  [11]Salz,Peggy Anne, Monitoring mobile app performance [J],dddmp,2014,9  [12]陈亚伟,王小丽,宋亚魁,郑明慧, 基于LoadRunner的移动端软件性能测试研究与应用[J],信息通信,2017 | | | |

|  |
| --- |
| **二、研究内容和目标（说明课题的具体研究内容，研究目标和效果，以及拟解决的关键科学问题。此部分为重点阐述内容）（不少于2500字）**  1.研究目标和效果  该课题研究目标旨在开发出一套UI自动化测试系统，该系统能够提供比较完备的UI自动化测试，基于自动化测试生成正确的遍历信息的图形结构和覆盖面比较广并且定制化强的测试用例，并能将简要的测试结果反馈到该系统的用户个人中心，详细的测试报告可以选择发送到邮箱或者下载到用户本地。  该系统预期能在UI自动化遍历，自动生成测试用例以及测试报告等方面提供给用户更加细致、完备和针对性强的测试，将来在行业中推广后，会大大提升测试效率和准确率。  2.研究内容  2.1遍历算法  项目中目前的遍历算法思想主要借鉴appcrawler的遍历思维：将app假象成一个图，app中的各个控件为图中的顶点，对app的操作为图的有向边，那么对于app的遍历即为对于图的遍历，这里使用图的深度优先遍历来模拟对app的遍历。  先来回顾一下何为深度优先遍历？  给定图G的初态是所有顶点均未曾访问过。在G中任选一顶点v为初始出发点(源点)，则深度优先遍历可定义如下：首先访问出发点v，并将其标记为已访问过；然后依次从v出发搜索v的每个邻接点w。若w未曾访问过，则以w为新的出发点继续进行深度优先遍历，直至图中所有和源点v有路径相通的顶点(亦称为从源点可达的顶点)均已被访问为止。若此时图中仍有未访问的顶点，则另选一个尚未访问的顶点作为新的源点重复上述过程，直至图中所有顶点均已被访问为止。  设x是当前被访问顶点，在对x做过访问标记后，选择一条从x出发的未检测过的边(x，y)。若发现顶点y已访问过，则重新选择另一条从x出发的未检测过的边，否则沿边(x，y)到达未曾访问过的y，对y访问并将其标记为已访问过；然后从y开始搜索，直到搜索完从y出发的所有路径，即访问完所有从y出发可达的顶点之后，才回溯到顶点x，并且再选择一条从x出发的未检测过的边。上述过程直至从x出发的所有边都已检测过为止。此时，若x不是源点，则回溯到在x之前被访问过的顶点；否则图中所有和源点有路径相通的顶点(即从源点可达的所有顶点)都已被访问过，若图G是连通图，则遍历过程结束，否则继续选择一个尚未被访问的顶点作为新的顶点，继续遍历。  了解以上遍历思想，再来将上述思想应用到对于app的自动化遍历上：首先启动待测app，然后开始选中其中一个控件开始遍历（点击或者输入等），并把该控件的访问设置标志为true（当递归回到该层的时候会略过访问过的控件直接去遍历其它未遍历过的控件），app会跳转至另一个界面，然后再去递归遍历该页面控件，不断递归下去，直到递归到某一层的页面无法再去跳转其它界面，此时递归结束，返回上一层继续遍历该层的控件......使用深度优先遍历的好处在于能够将对所有控件的遍历都覆盖到。  目前该算法还存在一些未解决的问题，比如遍历控件的优先级还未得到良好的控制（可以优先遍历input之类的标签等），遍历到需要第三方登录的控件时可能会发生crash等情况。  优化算法：  1）解决遍历顺序的优先级问题  2）在获取页面pageSource时，程序耗时较大，考虑能否优化？  3）遍历到webview时，拟使用js脚本生成DOM树来获取控件的属性  4）定义元素的URI来代替xpath  5）随着项目的持续推进，遍历算法可能还需围绕遍历深度、遍历控件的信息保存等方面进行优化  2.2生成遍历过程的图形结构  图形结构是指，在UI自动化遍历过程中，将遍历到的控件的信息如路径、属性等，根据遍历的顺序生成一个有向图，作为一次正确的自动化遍历的比对模板。  该图形结构中的顶点对应遍历的控件，页面间的跳转关系对应每一条有向边，执行遍历算法，完成遍历后就会生成相应的UI遍历图形结构。在生成图形结构的同时，也可以将每一步操作后的页面状态留作截图。  生成图形结构的意义在于，该图形结构是一次正确的遍历所得到的结果，那么当app发生异常时，就能利用该图形结构中顶点之间的关系以及每一次操作后生成的截图来快速定位异常位置，方便开发人员的维护，同时提升了公司效率。  2.3生成测试用例  在生成图形结构以后，那么每一个控件的路径，对控件的操作等相关信息就可以轻松获取到，我们可以利用这些信息生成xpath，然后利用appium封装成相应的测试用例，利用生成的图形结构来当作每一条测试用例的断言，用来比较真实值和期望值之间是否相等。将生成的测试用例打包给用户后，用户可以自行亲历遍历过程，也可以选中自己感兴趣或者关注的某些功能着重进行测试。  2.4测试报告自动生成  测试报告的生成分两步进行  1）在测试平台还未构建起来之前，可以使用junit5+allure生成测试报告。junit5 新增了对动态测试的支持，可以在运行时动态创建测试并执行。通过动态测试，可以满足一些静态测试无法解的需求，也可以完成一些重复性很高的测试。比如，有些测试用例可能依赖运行时的变量，有时候会需要生成上百个不同的测试用例。这些场景都是动态测试可以发挥其长处的地方。动态测试是通过新的@TestFactory 注解来实现的。测试类中的方法可以添加@TestFactory 注解的方法来声明其是创建动态测试的工厂方法。这种动态测试可以随时加入测试过程中app的截图、页面dom和元素属性（xpath信息）等。最后使用maven-surefire-plugin插件自带report功能，可以生成默认的xml测试报告，与此同时allure工具可以把xml转换成比较简单、直观且样式绚丽的html测试报告，增强用户体验。  2）测试平台构建完成后，可以将上述测试报告打包到测试报告中，用户可以在相关栏目直接下载测试报告，形成一个完善的测试流程。  2.5自动测试平台的构建  在上述核心功能完成后，就要把这些功能融入到UI自动化测试系统中了。  该系统大致分为两部分：  1）jenkins持续集成和部署的环境  2）展示给用户的一个门户网站  首先，jenkins环境的作用就是持续监听用户有无上传apk的行为，当有用户上传apk时，jenkins任务开始执行，安装apk，遍历app，生成图行结构，生成测试用例，并把这些结果打包发送给门户网站的相关模块。  其次是门户网站的设计与开发，拟使用springboot作为后台的开发工具，前端使用html+js。门户网站除了要有宣传企业的功能外，还要提供用户上传apk和接收测试结果的模块。测试结果在该系统中，可以只呈现核心的测试结果，详细的测试结果将会以某种格式（目前还未确定）发送到用户的邮箱。  3.拟解决的关键科学问题  3.1遍历算法的优化  1）目前定位app中的控件是采用元素的xpath信息，可能会存在定位不到元素的问题，因此现在考虑定义元素URI代替xpath：将能够唯一标识元素的属性组装成一条路径，这样在定位元素时就能准确找到控件。  2）遍历过程中如果遇到input之类的标签可以考虑优先遍历，而不是一开始就进行地毯式的遍历，因为往往需要输入的的控件出现问题的概率会大一点。除此之外，使用优先遍历后，遍历结果会更有层次，主次更加清晰。  3）遍历到webview时，采用一般的方法是获取不到控件的，因为webview是一种内嵌到安卓里html页面，所以针对这种元素拟使用js脚本生成DOM树，然后解析DOM树来获取控件的相关信息。  3.2生成图形结构  1）由于图形结构中要存储的信息比较杂乱，比如顶点要存储的信息包括该控件的resource-id/text/content-desc/location以及该元素的父元素的resource-id等属性，所以目前还未确定一个统一的节点类来存储控件信息。  2）遍历过程中每一步操作对应的截图是否应该存放在节点类当中，或者说可否仅用截图来额外生成一个UI遍历过程的图形结构，用于辅助测试人员更加直观的找出遍历出现异常的页面。 |

|  |
| --- |
| **三、研究方案设计及可行性分析（包括：研究方法，技术路线，理论分析、计算、实验方法和步骤及其可行性等）（不少于800字）**  1.研究方法  1）调查法  通过互联网查找进行调研，了解当下行业现状，以及目前企业用到的主要测试工具与测试手段，并了解行业发展前景以及将要落实的测试技术与测试方案。  2）文献研究法  通过阅读测试领域的相关文献，了解国内外测试领域的发展趋势与发展现状，以及正在酝酿中的测试方案。  3）比较研究法  调研了国内几家比较大的企业中关于测试领域的投入与研发，对比了各大企业之间的竞品的测试效果。  2.技术路线  1）提出课题并研究现状  提出UI自动化测试，并改进行业中目前存在的问题。行业中目前app测试的平台中比较成熟的仍然使用基于monkey的测试，更加深入的测试平台并没有成型，行业内迫切需要一套成型的并且测试效果更加完善的平台来提高测试效率。  2）建立理论基础  目前对于UI自动化测试的设计与开发可以基于appium和appcrawler，与此同时查阅国内外相关文献资料，参照行业内相关测试平台，去粗取精。  3）研究分析  通过调研国内外的测试平台，剖析其优势和缺点，既要看到其领先行业的内容，也要关注其局限性，这样在设计和开发UI自动化遍历系统时能统筹全局，避免开发过程中反复修改。  4）方案设计  针对本课题要研究的课题以及要实现的系统设计与开发，制定实现方案，确定系统架构和模块划分，设计用户交互界面。该系统后台计划使用springboot+jenkins，来实现测试报告的生成，使用html+js+css来实现系统用户之间的交互。  5）总结  评估最终的效果并落实设计与开发。  3.可行性分析  1）由于需要实时监听有无用户上传apk的行为，所以需要一个持续集成和部署的环境，所以可以部署jenkins环境。这样一来，当有用户上传apk时，jenkins服务开始执行，将执行结果反馈给测试平台。  2）执行结果需要展示给用户，因此可以采用springboot+html+js维护一个类似管理平台的管理中心，用于展示公司动态，上传apk，接收测试报告。  3）对于app的自动化遍历算法可以看成是对图的深度优先遍历，基于图的深度优先遍历可以保证将app中的每一个控件遍历完全，不会落下未测试的控件。  4）生成图形结构时，根据遍历结果，将遍历中获取到的控件信息统一格式后存储到节点类当中，对控件的操作信息存储到有向边中。  5）生成测试用例时利用生成的图形结构，在加上appium框架，可以轻易的组装/封装，生成一条条测试用例。  4.系统的体系架构  1）体系结构模式的选择  UI自动化遍历测试系统会有很多潜在用户，功能相对复杂，存储的信息量较大，需要专门的技术人员维护和管理系统，在体系结构的模式选择中，在满足安全与稳定的条件的同时，应尽量使管理维护操作简单。C/S就够体系过于庞大，管理维护复杂；P2P体系结构虽然功能强大，但是本系统并不需要即时通讯和不间断的数据更新；为使用户能够在简单的可视化界面下轻松方便的获取关心的数据，该系统采用B/S体系结构。  2）系统主要模块设计  模块用户提交apk自动化遍历apk，生成测试报告用户获取测试报告用户反馈测试结果  功能简介该系统提供一个上传待测apk的入口，可以触发jenkins任务jenkins任务触发后，后台会执行相应的遍历测试，并生成测试结果当系统整理话测试结果后，会告知用户去下载测试报告，用户可以下载到本地如果测试结果有问题，用户可以将问题反馈给平台  3）软硬件要求  该系统基于B/S体系结构，用户界面主要事务逻辑完全在服务器端通过浏览器实现，客户端一般的硬件配置均能满足要求，网络也不必是专门的网络硬件环境，但应用服务器运行数据负荷较重，需要更加优化的系统结构和相应硬件配置。  4）系统开发投入  B/S体系结构系统开发的投入与用户的多少无关，部署代价比较小，适合该UI自动化遍历系统的设计与开发。  5）维护与功能扩展  该系统基于B/S体系结构，所以只需维护服务器，所有的客户端只是浏览器，不需要任何维护和管理，而且只需将服务器连接专网，即可实现远程维护、升级和共享。  6）安全与稳定  B/S建立在广域网之上，面向不可知的用户群，所以该系统对安全的控制能力相对弱一点，比如受到攻击可能会泄露一些信息，这方面的问题无法根治，所以只能通过加密手段使安全能力得到优化提升。 |

|  |
| --- |
| **四、本研究课题可能的创新之处（不少于500字）**  与现有的测试平台相比，本课题可能存在的创新之处在于：  1.遍历算法  目前国内成型的测试平台大多都是基于安卓自带的monkey测试，这种测试的原理是利用socket通讯的方式来模拟用户的按键输入，触屏输入，手势输入等，看设备多长时间会出现异常，其主要作用是对安卓系统进行压力测试，测试app稳定性和健壮性。而该课题旨在研究出一套app的遍历算法，基于图的深度优先遍历，能遍历到app中的每一个控件，测试app运行的完整性。  2.图形结构  将遍历中的控件的相关信息存储成统一数据结构的节点并作为图的定点，对控件进行的操作/动作存储到有向边，这样就形成了UI的图形结构。目前在国内行业中并没有出现图形结构的生成，可以说是一个大胆的创新。  3.可定制的测试用例  根据生成的图形结构，我们可以比较轻松的获取到控件的相关信息，将其整理后封装成xpath路径，利用appium整合成相应的测试用例，并将这些测试用例发送给用户，用户可以选择关注的功能模块自行运行测试用例，复现测试过程，也能支持用户在不同机型上做兼容性测试。  4.性能测试  在UI自动化测试过程中，我们可以顺便获取到app执行过程中各个时段对于CPU的占用分析、内存使用情况、流量消耗，电量使用状态，以及运行过程中app的稳定性整理成柱状图或者折线图，同时也可以支持多机型之间的横向对比，以及同机型之间不同时段的纵向对比。 |
| **五、研究基础与工作条件（1.与本项目相关的研究工作积累基础 2.包括已具备的实验条件，尚缺少的实验条件和拟解决途径）（不少于500字）**  1.相关研究工作积累基础  本人从本科阶段到研究生阶段一直是软件工程专业的学生，一直致力于软件开发领域的学习和工作，熟悉集中常用的编程语言，开发框架，对于B/S或者C/S结构的设计与实现，以及前后端之间的通信与交互有比较多的经验。与此同时，由于要开发的这套系统是有关测试领域的，所以在开始项目之前，我又学习了一些主流的测试框架（appium）和测试工具（appcrawler）的使用，对于已经在开发中的这个移动应用的UI自动化测试系统的设计与实现有足够的信心。  2.实验条件  1）技术方面可以基于appium，在appcrawler的基础上改善遍历算法，整体模式的实现可以参照百度MTC、腾讯WeTest和Testin的实现。对于持续集成和部署部分可以使用jenkins环境，后台与前端的设计利用springboot+html+js实现。  2）知识方面可以多深入了解一下测试领域的现状以及发展趋势，多阅读一些国内外权威论文，总结并归纳相关测试知识。  综上，当前要做好技术储备（jenkins环境的配置与使用、springboot构建后台系统、html+js设计前端页面以及前后端交互以及使用java8开发UI自动化遍历的核心算法等），并持续不断得了解测试技术。  3.尚缺少的实验条件以及解决途径  由于开发使用的是windows系统，而在开发过程中jenkins中要执行的是shell脚本，因此为了方便后期开发，可能需要非windows机器。目前在电脑上安装了Cygwin64 Terminal，可以在windows上执行shell脚本。 |

**学位论文工作计划**

|  |  |  |
| --- | --- | --- |
| 时间 | 研究内容 | 预期效果 |
| 2018.09-2018.12 | 课题调研，准备开题 | 完成开题报告 |
| 2019.01-2019.03 | 调研行业需求，确定工作目标，攻克UI遍历算法 | 完成需求分析，完善并确定UI遍历算法 |
| 2019.04-2019.07 | 落实工作计划，保存遍历过程中的信息 | 完成概要设计和详细设计，生成正确的图形结构 |
| 2019.08-2019.10 | 测试报告的生成和系统整体的实现 | 生成测试用例，并结合之前的图形结构等生成一份比较完备的测试报告，并开发出人机交互良好的测试系统 |
| 2019.11-2020.02 | 对系统进行验证及改善，整理相关文档 | 基本完成论文的书写 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 评  定  小  组  成  员 | 姓 名 | 职 称 | 单位名称 | 职务 |
| 漆涛 | 教授 | 北京邮电大学 | 成员 |
| 陈莉萍 | 讲师 | 北京邮电大学 | 成员 |
| 雷友珣 | 副教授 | 北京邮电大学 | 成员 |
| 崔毅东 | 副教授 | 北京邮电大学 | 成员 |
| 杨谈 | 副教授 | 北京邮电大学 | 成员 |
|  |  |  |  |
| 导师意见： | | | | |
|  | | | | |
| 导师（签名）：  日期： 年 月 日 | | | | |
| 开题报告小组意见： | | | | |
| 组长（签名）：  日期： 年 月 日 | | | | |
| 学院意见（签章）： | | | | |
| 负责人：  日期： 年 月 日 | | | | |