

文章编号: 2095-2163(2019)02-0159-03

中图分类号: TP311.53

文献标志码: A

基于 TestLink 的软件测试自动化管理

王妍

(曲阜师范大学 软件学院, 山东 曲阜 273165)

摘要: 软件测试是保证软件产品质量的必要手段和方法。文章从自动化软件测试管理的角度出发,解释了自动化软件测试的过程和任务,存在的问题以及解决办法。并介绍了如何用开源软件 TestLink 对在线考试系统进行自动化的测试过程管理,并自动产生了测试结果及测试报告。

关键词: 软件测试; 软件自动化测试; TestLink; 测试管理

The automation management of software testing based on TestLink

WANG Yan

(School of Software Engineering, Qufu Normal University, Qufu Shandong 273165, China)

[Abstract] Software testing is the necessary means and methods to ensure the quality of software products. From the point of view of automatic software testing management, the paper explains the process and task of automated software testing, as well as the existing problems and the corresponding solutions. After that, the paper introduces how to use the open source software TestLink to automate the testing process management of the online examination system, and automatically generate the test results and test reports.

[Key words] software testing; software automation test; TestLink; test management

0 引言

软件测试是从测试立项工作开始的,然后详拟测试计划,分配测试任务,执行测试到测试完结生成质量报告,这一过程称为软件测试过程。自动化测试管理可以大大简化这一复杂的过程,一般需要对测试需求分析、测试计划制定、测试设计与开发、测试执行与缺陷跟踪、测试评估等5个阶段进行有效管理。对此可做研究分述如下。

1 自动化测试的基本任务

1.1 确定测试需求

测试活动总是从研究需求分析开始的,需求分析是开展测试活动的基础和依据,软件需求分析可以告诉研究者需要“测什么”。测试人员需要将测试与需求对应起来,以便为之后的测试情况确定测试需求覆盖率。测试需求分析阶段要求测试人员首先要理解并重构系统的业务流程;然后对系统的具体功能需求、性能需求以及安全性需求进行有效分析;最后对界面的美观性及易操作性进行分析。

1.2 制定测试计划

测试计划是测试者用来执行测试用例的依据,

测试计划由测试用例组成。测试计划要注重过程的安排,从编写测试计划过渡到计划测试任务,重点是详尽的计划安排。在 TestLink 系统中,一个完整的测试计划要包括各阶段测试的名称。

1.3 设计和生成测试用例

测试用例是指测试过程中为了发现缺陷而向系统提供的数据、操作及预期结果。测试用例是执行测试的最小实体,是软件在某个具体的测试场景下能够正常执行并达到预期结果的过程。TestLink 支持2层测试用例管理:测试用例集、测试用例,另外还可以提供管理测试需求与测试用例对应关系的功能。

1.4 测试执行及缺陷记录

执行测试用例是为了发现软件缺陷。执行测试用例的过程,实际上是对比实际输出结果和预期结果是否一致,凡不一致的作为软件缺陷处理。TestLink 可以集成 Mantis 对软件缺陷进行自动化的管理,可以对大量原始测试数据进行有效管理,另外对于发现的缺陷,可以实现跟踪管理和监控,直到缺陷被修改并且关闭。

1.5 测试结论

测试结果分为4种情况:通过、失败、锁定、尚未

基金项目: 山东省高校科技计划项目(KJ2018BBN101); 曲阜师范大学科技计划项目(xkj201620); 曲阜师范大学实验技术项目(XN201702)。

作者简介: 王妍(1980-),女,硕士,实验师,主要研究方向:软件测试。

收稿日期: 2018-12-18

执行。整个测试过程结束后,可以通过测试覆盖率、测试执行率、测试执行通过率、测试缺陷解决率等指标来对测试结果进行评估。TestLink 根据测试过程中记录的数据,提供了多种度量统计功能,可以从测试进度、失败的测试用例、锁定的测试用例及尚未执行的测试用例等几个方面进行分析总结。

2 TestLink 测试流程

2.1 TestLink 中的角色分工

在 TestLink 系统中,每个用户都可以维护自己的私有信息,初次登录可以使用 admin 用户名,然后可

表 1 TestLink 用户角色及权限

Tab. 1 User roles and permissions of TestLink

| 角色 | 权限列表 | 能力 |
|---------------|---------------------|-------------|
| Guest | 查看测试用例 | 浏览数据 |
| Test Executor | 执行测试用例 | 执行测试 |
| Test Analyst | 创建、查看、执行、测试用例 | 编辑测试规范及执行需求 |
| TestDesigner | 以上所有权限及修改测试需求 | 编辑测试规范及测试需求 |
| TestLeader | 以上所有权限及管理测试进度、编辑里程碑 | 编辑测试规范及测试需求 |
| Administrator | 以上所有权限 | 一切可行性的操作 |

3 TestLink 自动化测试管理

3.1 TestLink 介绍

TestLink 是一款开源的软件自动化测试软件,采用 PHP 开发,是一个 B/S 模式的测试管理平台。该平台可以对测试流程的各个阶段进行有效的组织和管理,包括创建测试脚本、制定测试计划、执行测试和跟踪测试结果,还可以动态收集和组织测试用例,根据测试用例覆盖率和通过率产生测试报告及图表。其主界面如图 1 所示。

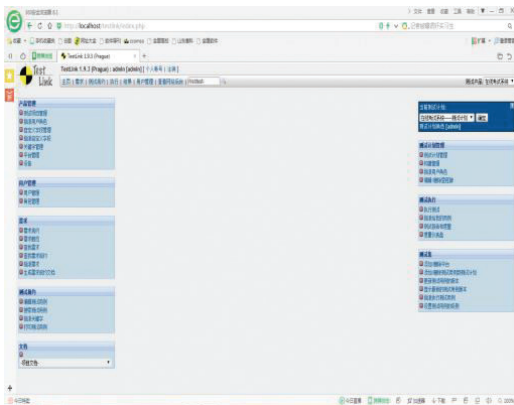


图 1 TestLink 主界面

Fig. 1 Main interface of TestLink

该软件的优点是提供 Web 访问方式,用户可以

以创建新用户。TestLink 提供了 6 种不同的权限级别,不同的权限级别可以工作的权限不同,详见表 1。

2.2 TestLink 测试流程

TestLink 可以对整个测试过程进行合理的管理,一般从创建项目开始,然后创建测试需求,创建测试用例,测试计划与测试需求连接,为需求指派用例,在计划中添加测试用例,指派分配测试任务到不同角色,执行测试,分析测试结果。整个流程采取自上而下的顺序执行,前一步是后一步的前提,后一步是前一步的结果。

自定义角色,测试可以根据优先级指派测试员,还可以自定义里程碑等等。

3.2 TestLink 对在线考试系统进行自动化测试管理

3.2.1 测试需求管理

在线考试系统是一个 B/S 架构的系统,分为前台操作和后台管理两个大模块,分析其业务流程,分为登录验证、学生注册、成绩查询、在线考试、答题、查询成绩共 6 个子模块。通过 TestLink 的需求规约管理,建立需求规格树,如图 2 所示。

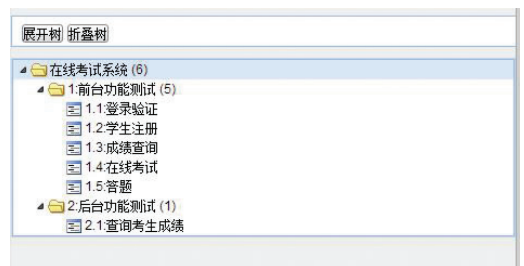


图 2 TestLink 需求规约树

Fig. 2 Requirements specification tree of TestLink

3.2.2 测试用例管理

根据在线考试系统的需求规约设计测试用例,其中前台功能测试需要 3 个测试用例,在线考试需要 3 个测试用例,后台功能测试需要 1 个测试用例,创建好的测试用例树如图 3 所示。



图 3 测试用例树

Fig. 3 Test case tree

测试用例和系统需求之间是 n 对 n 的关系。TestLink 允许在需求和测试用例之间建立关系, 一个需求可以被指派给零个或多个测试用例上, 反之亦如此。

3.2.3 测试计划管理

在 TestLink 中可以通过测试计划管理对测试的过程进行有效管理。测试计划一般包括计划名称、计划描述、计划开始时间、计划预计完成时间、测试用例状态描述、版本信息等内容。TestLink 提供了测试计划创建、角色分配、测试用例分配、测试用例添加、删除等操作。

3.2.4 测试结果分析

TestLink 可以根据测试过程中记录的数据, 进行多种分析和度量, 可以自动生成测试计划报告、测试报告、测试总体计划进度。通过 TestLink 对在线考试系统的测试过程进行管理, 共指派 8 个测试用例, 测试需求覆盖率 100%, 通过率 74.8%, 其中锁定比例 12.6%, 失败率 12.6%。

另外, TestLink 还可以生成图表形式等比较直观的报告, 以在线考试系统为例, 可以产生测试结果的饼状图, 如图 4 所示。

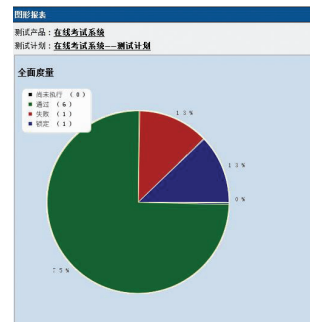


图 4 总体测试结果饼状图

Fig. 4 Pie chart of overall test results

4 结束语

由此可见, TestLink 已经创造了良好的测试管理条件, 其配置及使用都非常方便, 并且是开源的测试工具, 基于 B/S 架构的模式使得其操作更加灵活。但是 TestLink 也存在一些缺点, 比如返回需要重新链接, TestLink 和缺陷管理工具需要手工整合等等。就测试过程的自动化管理效果来看, TestLink 可以有效管理测试过程, 分角色权限管理软件测试, 是一个非常好的软件测试管理平台。

参考文献

- [1] 王蓁蓁. 软件测试理论初步框架[J]. 计算机科学, 2014, 41(3): 12-16, 35.
- [2] 曹鹤玲, 姜淑娟, 鞠小林. 软件错误定位研究综述[J]. 计算机科学, 2014, 41(2): 1-6, 14.
- [3] 王蓁蓁, 徐宝文, 周毓明, 等. 一种随机 TBFL 方法[J]. 计算机科学, 2013, 40(1): 5-13, 18.
- [4] 韩丽霞. 求解多目标优化问题的新遗传算法[J]. 计算机科学, 2013, 40(6): 64-66, 95.
- [5] 蔡建平, 倪建成, 高仲合. 软件测试实践教程[M]. 北京: 清华大学出版社, 2014.
- [6] 周百顺, 张伟, 陈良臣. 应用软件测试实践[M]. 北京: 清华大学出版社, 2014.

(上接第 158 页)

- [7] GENG Zhe, DENG Hai, HIMED B. Adaptive radar beamforming for interference mitigation in radar-wireless spectrum sharing[J]. IEEE Signal Processing Letters, 2015, 22(4): 484-488.
- [8] YANG Yunchuan, SUN Cong, ZHAO Hui, et al. Algorithms for secrecy guarantee with null space beamforming in two-way relay networks[J]. IEEE Transactions on Signal Processing, 2014, 62(8): 2111-2126.
- [9] 郭华平, 董亚东, 毛海涛, 等. 一种基于逻辑判别式的稀有分类方法[J]. 小型微型计算机系统, 2016, 37(1): 140-145.
- [10] XU Li, JIA Jiaya, MATSUSHITA Y. Motion detail preserving optical flow estimation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 34(9): 1744-1757.
- [11] LIU Zhi, ZHANG Xiang, LUO Shuhua, et al. Superpixel-based

spatiotemporal saliency detection [J]. IEEE Transactions on Circuits and Systems for Video Technology, 2014, 24(9): 1522-1540.

- [12] 张红蕊, 张永, 于静雯. 云计算环境下基于朴素贝叶斯的数据分类[J]. 计算机应用与软件, 2015, 32(3): 27-30.
- [13] 吴志军, 李光, 岳猛. 基于信号互相关的低速率拒绝服务攻击检测方法[J]. 电子学报, 2014, 42(9): 1760-1766.
- [14] 张博, 郝杰, 马刚, 等. 混合概率典型相关性分析[J]. 计算机研究与发展, 2015, 52(7): 1463-1476.
- [15] 贾昊, 董泽, 闫来清. 基于信号分解和统计假设检验的稳态检测方法[J]. 仪器仪表学报, 2018, 39(10): 150-157.
- [16] 蒋帅, 孙小凡, 向茂生, 等. 一种基于 DEM 的机载干涉相位生成算法[J]. 国外电子测量技术, 2018, 37(9): 56-61.