

黑盒测试设计与执行

软件质量保障与测试课程 Lab5 课程作业（第 9 组）

Tian, Jiahe^{*} Hu, Xiaoxiao[†] Huang, Jiani[‡] Liu, Jiaking[§]
Shi, Ruixin[¶] Wu, Chenning^{||} Zhang, Cenyuan^{**}
Zhang, Yihan^{††} Wang, Chen^{‡‡}

2020 年 5 月 2 日

^{*}Equal Contribution, Fudan University, 17307130313 (tianjh17@fudan.edu.cn)

[†]Equal Contribution, Fudan University, 17302010077 (xxhu17@fudan.edu.cn)

[‡]Equal Contribution, Fudan University, 17302010063 (huangjn17@fudan.edu.cn)

[§]Equal Contribution, Fudan University, 17302010049 (jiakingliu17@fudan.edu.cn)

[¶]Equal Contribution, Fudan University, 17302010065 (rxshi17@fudan.edu.cn)

^{||}Equal Contribution, Fudan University, 17302010066 (cnwu17@fudan.edu.cn)

^{**}Equal Contribution, Fudan University, 17302010068 (cenyuanzhang17@fudan.edu.cn)

^{††}Equal Contribution, Fudan University, 17302010076 (zhangyihan17@fudan.edu.cn)

^{‡‡}Equal Contribution, Fudan University, 16307110064 (wangc16@fudan.edu.cn)

黑盒测试设计与执行

软件质量保障与测试课程 *Lab5* 课程作业

摘要

本次作业为软件质量保障与测试课程的 Lab5 课程作业，需要我们以小组为单位完成对出题系统的黑盒测试。本文档分为三小节。第一小节介绍了本小组设计的黑盒测试整体框架以及测试流程；第二小节介绍了黑盒测试中的关键字以及测试数据设计；第三小节介绍了本小组对黑盒测试任务的具体脚本实现以及运行情况介绍。

关键词

系统与软件工程; 系统与软件质量要求和评价; 测试文档

目录	3
----	---

目录

摘要	2
关键词	2
1 测试框架设计	4
1.1 黑盒测试框架图	4
1.2 黑盒测试框架图说明	9
2 关键字及测试数据设计	9
2.1 测试关键字	9
2.1.1 登录	9
2.1.2 创建考题	10
2.2 测试数据	10
2.2.1 登录	10
2.2.2 创建考题	12
3 测试脚本实现及运行	17
3.1 脚本实现方式	17
3.1.1 出题测试脚本实现	17
3.2 运行截图及说明	17
3.2.1 运行说明	17
3.2.2 控制台截图	19
3.3 测试用例运行过程记录	19
参考文献	20

1 测试框架设计

1.1 黑盒测试框架图

在本节内容中，将会通过两个图对本小组设计的黑盒测试框架进行描述。

其中程序流程图将整个黑盒测试的各个步骤进行了分解，展现了整个测试过程中不同动作执行的先后顺序以及之间的关联关系。而 UML 类图将黑盒测试的框架组成进行了分解，说明了测试框架中各个模块之间的关系。

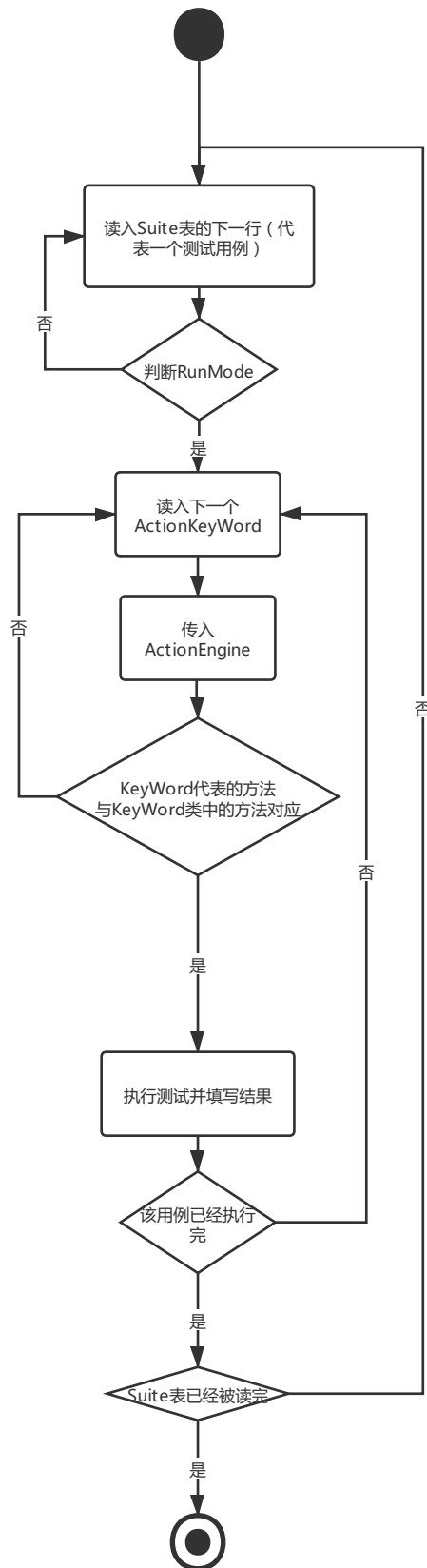


图 1: 黑盒测试流程图

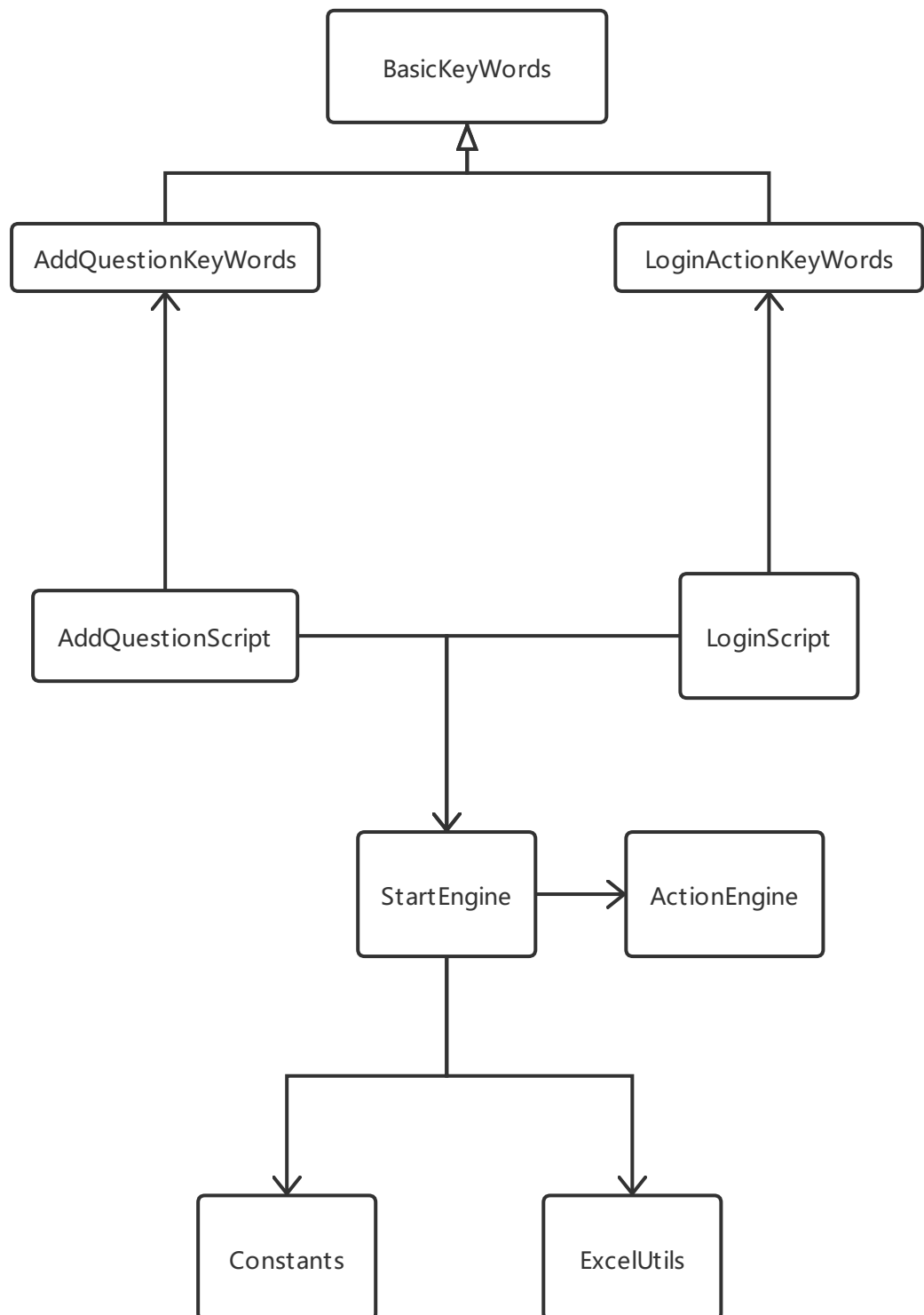


图 2: 黑盒测试整体框架图

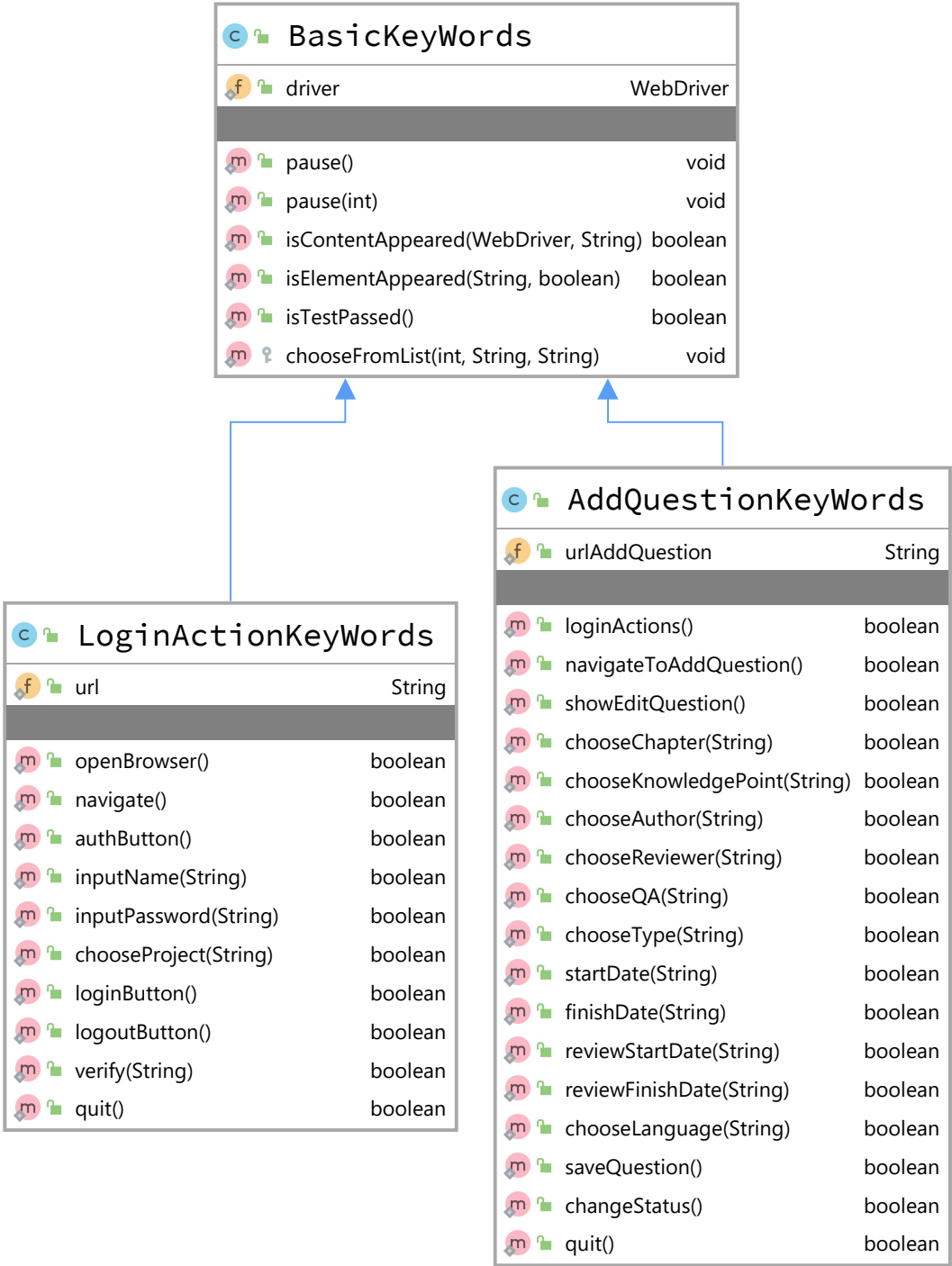


图 3: 黑盒测试关键词设计框架图

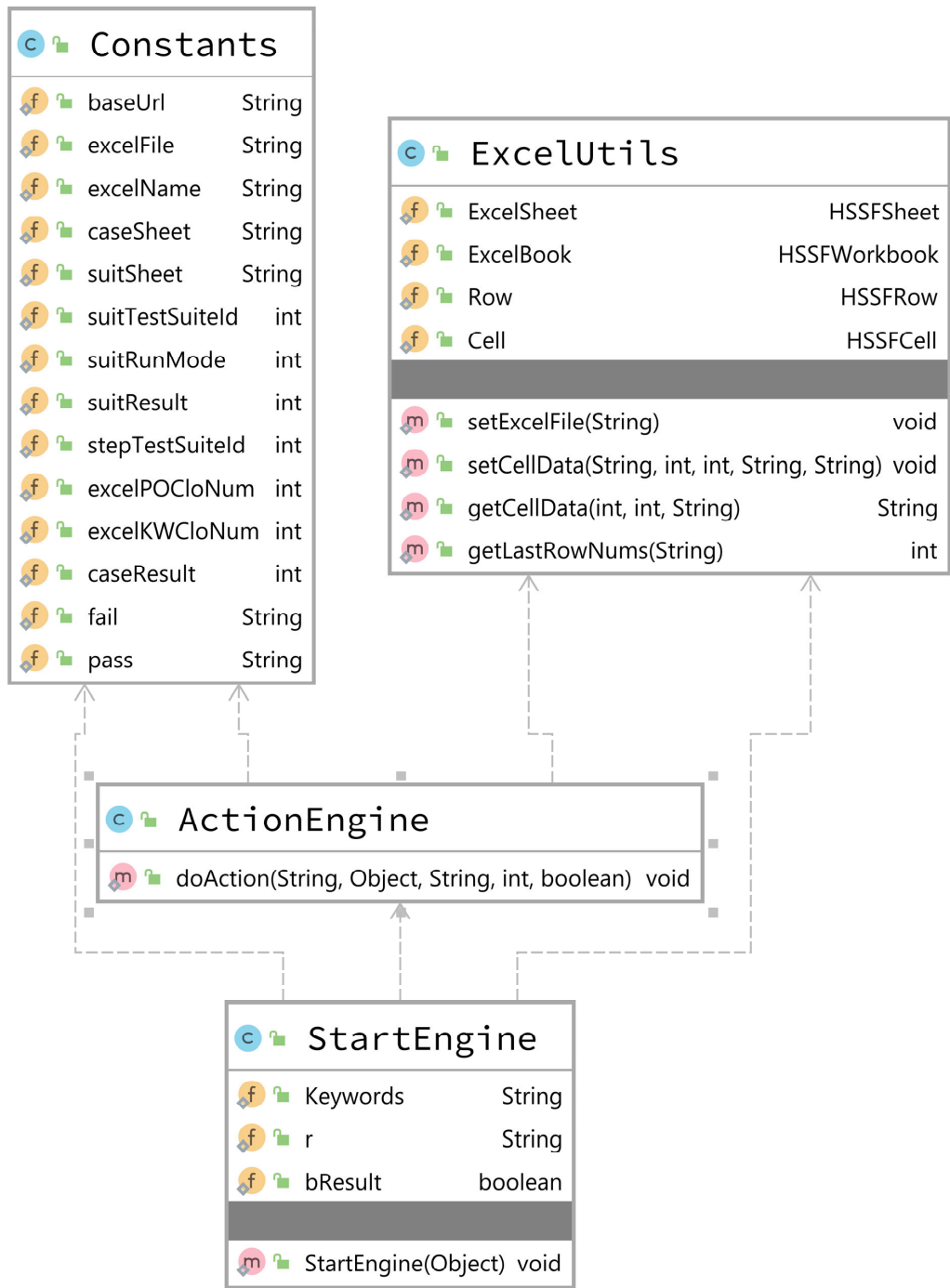


图 4: 黑盒测试用例执行组件框架图

1.2 黑盒测试框架图说明

本小节内容对本小组所做的黑盒测试框架进行说明，同时也是对上一小节中的框架图的进一步说明。

框架主体结构有 *config*, *script*, *util* 三部分。

config 部分中的类(LoginActionKeywords, AddQuestionKeywords, 它们都继承具有通用方法(如开始, 结束, 寻找页面元素)的 BasicKeywords)是测试场景的关键字汇总, 需要测试的测试用例都能够拆分为类中的关键字, 通过关键字的组合来实现测试用例的执行。

script 部分是测试脚本的执行入口, 通过调用 Engine 类来开始测试。

util 部分包含工具类和 Engine 类。工具类为读入 Excel 文件的 ExcelUtils, 记录需要使用的常量(如数据文件路径, 记录测试用例的表名称等)的 Constants, 以及 StartEngine 和 ActionEngine。

StartEngine 的入口方法以代表场景的 Keyword 类的实例作为参数, 循环读取 suitSheet 里面的值, 找出运行的场景, 找到运行场景后根据 stepTestSuiteId 在 caseSheet 中循环查找相对应的执行步骤。

对于 caseSheet (也即一个测试用例)的具体某行(也即场景中的某个关键字), 则调用 ActionEngine 来执行, ActionEngine 检查该行的关键字是否对应关键字类中的一个方法, 如果是则调用该方法并传入可能有的参数, 进行代表该关键字的步骤的执行。

2 关键字及测试数据设计

2.1 测试关键字

2.1.1 登录

登录功能需要完成用户登录出题系统的功能。登录包括打开浏览器、导航到登录网页、输入用户名、输入密码、点击认证、选择项目、登录、登出、退出。于是我们针对上述的关键步骤设计了以下关键字：

- openBrowser 打开浏览器
- navigate 页面导航
- inputName 输入用户名
- inputPassword 输入密码
- authButton 点击认证
- chooseProject 选择项目
- loginButton 登录

- logoutButton 登出
- quit 退出

2.1.2 创建考题

创建考题需要完成考题的相关设定。这些设定包括选择章节、选择知识点、设置作者、设置评审、设置质管、设置类型、设置语言、设置出题日期、设置评审日期。而创建考题作为一个独立的 UI 测试，登录是一个基础环节。于是我们针对上述的关键步骤设计了以下关键字：

- loginActions 登录系统
- navigateToAddQuestion 页面导航
- showEditQuestion 选择添加考题
- chooseChapter 选择章节
- chooseKnowledgePoint 选择知识点
- chooseAuthor 设置作者
- chooseReviewer 设置评审
- chooseQA 设置质管
- chooseType 设置题目类型
- startDate 设置出题开始日期
- finishDate 设置出题结束日期
- reviewStartDate 设置评审开始日期
- reviewFinishDate 设置评审结束日期
- chooseLanguage 选择语言
- saveQuestion 保存考题

其中我们没有将登录拆分为更加细致的行为，这是因为登录对于创建考题环节而言是比较基础和简单的。它不是创建考题黑盒测试的考察内容。

2.2 测试数据

2.2.1 登录

登录的 UI 测试分为两个阶段，一个是用户名密码认证。如果认证成功可以进行项目选择登录。

2.2.1.1 等价类划分

按输入分析等价类

按输出分析等价类 (先认证, 认证成功会输出项目列表)

表 1: 等价类划分-输入

输入参数	有效等价类	无效等价类
username	[1] 已注册的用户名, 且用户名下有项目, 用户是 enabled 状态	[2] 已注册的用户名, 用户名下无项目, 用户是 enabled 状态 [3] 用户名为空或 null [4] 未注册的非 null 非空的用户名 [5] 已注册的用户名, 但是用户被禁用了
password	[6] 用户设置的密码, 与用户名相对应	[7] 用户名正确, 但密码错误 [8] 用户名未注册
project_id	[9] 项目已存在, 且用户参与	

表 2: 等价类划分-输出

输入参数	有效等价类	无效等价类
project_list	[10] 项目列表正确 (符合要求)	[11] 项目列表错误 (增漏或不符合要求)

表 3: 等价类划分-字符串本身

输入参数	有效等价类	无效等价类
大小写	[12] 字符串全小写 [13] 字符串全大写 [14] 字符串大小写混杂	

从字符串本身分析

由于一些限制, 从前端输入的时候有些情况是做不到的. 比如将字符串设置为 null, 选择不存在的项目.

从已注册用户角色分析

表 4: 等价类划分-角色

输入参数	有效等价类	无效等价类
是否为管理员	[15] 管理员 [16] 非管理员	
选择项目中用户的角色	[17] 主持人 [18] 主持人以外的参与者	

边界值分析适用于具有连续取值的参数分析, 登录输入没有连续取值所以不存在边界值分析。

2.2.2 创建考题

2.2.2.1 等价类划分

创建考题的 UI 测试具备多个输入参数, 并且这些参数的取值也多种多样。我们对输入参数进行等价类划分:

表 2 是根据等价类划分设计的测试用例, 其中 c0 表示第一个章节, k0 表示该章节第一个知识点, u0 表示成员 1, u1 表示成员 2, u2 表示成员 3, t0 表示类型为情景题, l0 表示语言为中文。

表 5: 等价类划分测试数据

ID	覆盖的类	username	pwd	pj	Expect output	描述
1	[1][6][9][10] [12][15][17]	testadmin	test	3	List:test0316 test0316b 项目测试测试登陆专用项目 成功登陆	主持人成功登陆项目
2	[1][6][9][10] [12][16][18]	jmeter004	123456	1	list:test0316 成功登陆	非主持人成功登陆项目
3	[1][6][9][10] [12][13][15][17]	TESTADMIN	test	1	List:test0316 test0316b 项目测试测试登陆专用项目 成功登陆	检查用户名是否大小写敏感
4	[1][6][9][10] [12][14][15][17]	TeSTADmIN	test	1	List:test0316 test0316b 项目测试测试登陆专用项目成功登陆	检查用户名是否大小写敏感
5	[1][6][11][12] [13][15][17]	testadmin	TEST		登陆失败, 用户名或密码错误!	检查用户密码是否大小写敏感
6	[1][6][11][13] [14][15][17]	testadmin	teSt		登陆失败, 用户名或密码错误!	检查用户密码是否大小写敏感
7	[2][6][11][12] [16]	jmeter005	123456		您暂时没有分配到任何项目, 或者您参与的项目已经结束. 如需帮助请联系管理员!	没有参与项目的人无法登陆
8	[3][8]				请输入用户名和密码!	不输入用户名和密码无法登陆
9	[4][8][12]	imnsb	123456		登陆失败, 用户名或密码错误!	未注册用户无法登陆
10	[5][6][14][16]	testLogin	123456		账户已被锁定, 如需帮助请联系管理员!	禁用用户无法登陆
11	[2][7][12][16]	jmeter004			请输入用户名和密码!	密码错误且密码为空
12	[2][7][12][14] [16]	jmeter004	Test		登陆失败, 用户名或密码错误!	密码错误且密码不为空

表 6: 等价类划分

输入参数	有效等价类	无效等价类
Chapter	[1] 任何章节	[2] 为空
Knowledge point	[3] 章节下任意知识点	[4] 为空
Author	[5] 任意用户	[6] 为空
Reviewer	[7] 除 author 外用户	[8]author [9] 为空
QA	[10] 除 author, reviewer 外用户	[11]author 或 reviewer [12] 为空
Type	[13] 任意类型	[14] 为空
Start date	[15] 任意日期	[16] 为空
Finish date	[17] 在开始日期及之后的日期	[18] 为空
Review start date	[19] 出题开始日期及之后日期	[20] 为空 [21] 出题开始日期之前
Review finish date	[22] 评审开始日期及之后的日期, 且不早于出题结束日期	[23] 为空 [24] 出题结束日期之前
Language	[25] 任意语言	[26] 为空

2.2.2.2 边界值分析

边界值分析适用于具有连续取值的参数分析, 题目中具有连续取值的只有出题日期与评审日期。其中出题开始日期不存在边界限定, 故不考虑。而对于评审结束日期, 它的边界值与评审开始日期和出题结束日期相关。由于这两个日期不存在约束关系, 故它的多个边界值条件可以同时成立。对于不存在连续取值的参数, 此处边界值分析不再进行罗列, 在测试数据中会使用固定的取值。

表 4 是根据边界值分析设计的测试数据。

表 7: 等价类划分测试数据

ID	类	输入参数											预期
1	1,3,5,7,10,13,15,17,19,22,25	c0	k0	u0	u1	u2	t0	2020-05-03	2020-05-03	2020-05-03	2020-05-03	10	成功
2	2	空	k0	u0	u1	u2	t0	2020-05-03	2020-05-03	2020-05-03	2020-05-03	10	失败
3	4	c0	空	u0	u1	u2	t0	2020-05-03	2020-05-04	2020-05-03	2020-05-04	10	失败
4	6	c0	k0	空	u1	u2	t0	2020-05-03	2020-05-04	2020-05-03	2020-05-04	10	失败
5	8	c0	k0	u0	u0	u2	t0	2020-05-03	2020-05-04	2020-05-03	2020-05-04	10	失败
6	9	c0	k0	u0	空	u2	t0	2020-05-03	2020-05-04	2020-05-03	2020-05-04	10	失败
7	11	c0	k0	u0	u1	u1	t0	2020-05-03	2020-05-04	2020-05-03	2020-05-04	0	失败
8	12	c0	k0	u0	u1	空	t0	2020-05-03	2020-05-04	2020-05-03	2020-05-04	10	失败
9	14	c0	k0	u0	u1	u2	空	2020-05-03	2020-05-04	2020-05-03	2020-05-04	10	失败
10	16	c0	k0	u0	u1	u2	t0	空	2020-05-04	2020-05-03	2020-05-04	10	失败
11	18	c0	k0	u0	u1	u2	t0	2020-05-03	空	2020-05-03	2020-05-04	10	失败
12	20	c0	k0	u0	u1	u2	t0	2020-05-03	2020-05-04	空	2020-05-04	10	失败
13	21	c0	k0	u0	u1	u2	t0	2020-05-03	2020-05-04	2020-04-30	2020-05-04	10	失败
14	23	c0	k0	u0	u1	u2	t0	2020-05-03	2020-05-04	2020-05-03	空	10	失败
15	24	c0	k0	u0	u1	u2	t0	2020-05-03	2020-05-04	2020-05-03	2020-05-03	10	失败
16	26	0	0	u0	u1	u2	t0	2020-05-03	2020-05-04	2020-05-03	2020-05-04	空	失败

表 8: 边界值分析

输入参数	边界值
Finish date	[27] 与 start date 相同 [28] start date 后一天
Review start date	[29]Start date 前一天 [30] 与 Start date 相同 [31]Start date 后一天
Review finish date	[32] 与 Review start date 相同 [33]Review start date 后一天 [34]Finish date 前一天 [35] 与 Finish date 相同 [36]Finish date 后一天

表 9: 边界值分析测试数据

ID	类	输入参数										预期	
17	27,30,32,35	c0	k0	u0	u1	u2	t0	2020-05-03	2020-05-03	2020-05-03	2020-05-03	10	成功
18	28,31,33,36	c0	k0	u0	u1	u2	t0	2020-05-03	2020-05-04	2020-05-04	2020-05-05	10	成功
19	29	c0	k0	u0	u1	u2	t0	2020-05-03	2020-05-04	2020-05-02	2020-05-04	10	失败
20	34	c0	k0	u0	u1	u2	t0	2020-05-03	2020-05-04	2020-05-03	2020-05-03	10	失败

3 测试脚本实现及运行

3.1 脚本实现方式

脚本的实现主要参照 <https://my.oschina.net/hellotest/blog/531932> 中，通过 java 反射机制，使用 poi 包操作 excel 关键字表，使用 selenium 的 jar 包进行自动化录制测试操作，以完成自动化测试框架。

3.1.1 出题测试脚本实现

3.1.1.1 执行流程

- 登陆
- 导航到添加题目
- 显示添加题目对话框
- 选择章节
- 选择知识点
- 选择作者
- 选择审核者
- 选择质量审核
- 选择种类
- 填写开始日期
- 填写结束日期
- 填写审核开始日期
- 填写审核结束日期
- 选择语言
- 保存问题

3.2 运行截图及说明

3.2.1 运行说明

测试用例 2 和测试用例 3 保存题目预期结果应该是失败，但由于系统存在 bug，并不会对章节和知识点的缺失进行检查，所以实际 UI 测试的结果并不是失败。

A	B	C	D
TestSuiteId	Description	Runmode	Results
login_001	登陆场景测试	NO	PASS
question_001	测试用例1	YES	PASS
question_002	测试用例2	YES	PASS
question_003	测试用例3	YES	PASS
question_004	测试用例4	YES	PASS
question_005	测试用例5	YES	PASS
question_006	测试用例6	YES	PASS
question_007	测试用例7	YES	PASS
question_008	测试用例8	YES	PASS
question_009	测试用例9	YES	PASS
question_010	测试用例10	YES	PASS
question_011	测试用例11	YES	PASS
question_012	测试用例12	YES	PASS
question_013	测试用例13	YES	PASS
question_014	测试用例14	YES	PASS
question_015	测试用例15	YES	PASS
question_016	测试用例16	YES	PASS
question_017	测试用例17	YES	PASS
question_018	测试用例18	YES	PASS
question_019	测试用例19	YES	PASS
question_020	测试用例20	YES	PASS

图 5: result screenshot

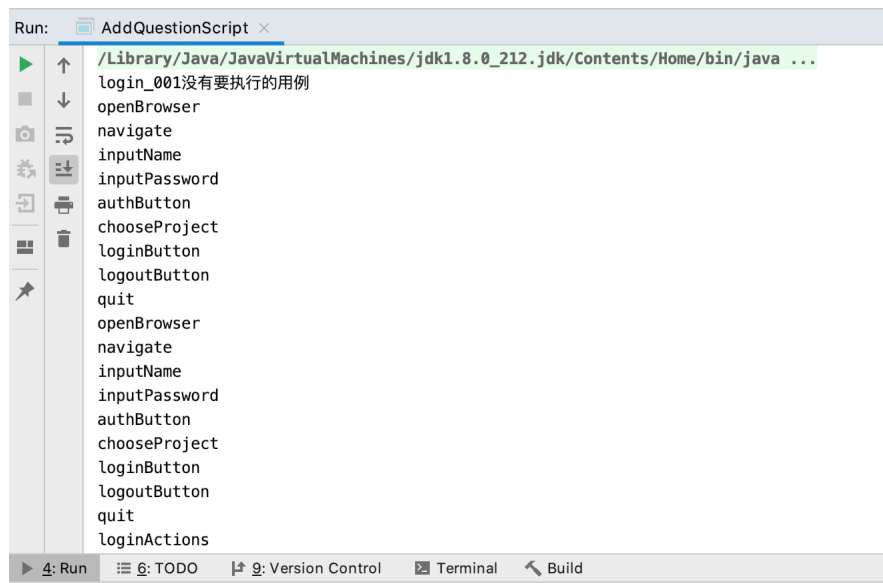


图 6: console screenshot

3.2.2 控制台截图

3.3 测试用例运行过程记录

本小组对本次黑盒测试的过程进行了记录，这些记录的内容完整地展现了本小组的黑盒测试用例设计以及测试环境、测试流程。测试过程视频可以通过如下方式进行查看，各平台的内容一致：

- 优酷视频（访问密码为 FudanUniversitySoftwareSchool）：https://v.youku.com/v_show/id_XNDY2NzE4NzQxMg==.html
- 搜狐视频（访问密码为 FudanUniversitySoftwareSchool）：<https://tv.sohu.com/v/dXMvMTI2MDUwODM0LzE5MzY2NDI2OS5zaHRtbA==.html>
- 百度网盘（提取码为 ut6f）：https://pan.baidu.com/s/1FI7FPEp3_piM6QpkU1B7WQ

参考文献

International Organization for Standardization. 2014. *Systems and Software Engineering — Systems and Software Quality Requirements and Evaluation (SQuaRE) — Guide to SQuaRE*. International Organization for Standardization. Vol. 2014. <https://www.iso.org/standard/64764.html>.

中国国家标准化管理委员会. 2016. *GB/T 25000.51-2016《系统与软件工程系统与软件质量要求和评价 (SQuaRE) 第 51 部分: 就绪可用软件产品 (RUSP) 的质量要求和测试细则》*. 系统与软件工程系统与软件质量要求和评价 (SQuaRE). Vol. 51. 中国国家标准化管理委员会. <http://openstd.samr.gov.cn>.

———. 2017a. *GB/T 25000.12-2017《系统与软件工程系统与软件质量要求和评价 (SQuaRE) 第 12 部分: 数据质量模型》*. 系统与软件工程系统与软件质量要求和评价 (SQuaRE). Vol. 12. 中国国家标准化管理委员会. <http://openstd.samr.gov.cn>.

———. 2017b. *GB/T 25000.24-2017《系统与软件工程系统与软件质量要求和评价 (SQuaRE) 第 24 部分: 数据质量测量》*. 系统与软件工程系统与软件质量要求和评价 (SQuaRE). Vol. 24. 中国国家标准化管理委员会. <http://openstd.samr.gov.cn>.

———. 2018. *GB/T 25000.40-201《系统与软件工程系统与软件质量要求和评价 (SQuaRE) 第 40 部分: 评价过程》*. 系统与软件工程系统与软件质量要求和评价 (SQuaRE). Vol. 40. 中国国家标准化管理委员会. <http://openstd.samr.gov.cn>.

———. 2019. *GB/T 25000.23-2019《系统与软件工程系统与软件质量要求和评价 (SQuaRE) 第 23 部分: 系统与软件产品质量测量》*. 系统与软件工程系统与软件质量要求和评价 (SQuaRE). Vol. 23. 中国国家标准化管理委员会. <http://openstd.samr.gov.cn>.