<小箱交云作业平台>

系统测试报告

版本 <1.1>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| <06/01/21> | <1.0> | 初稿 | 程可、邓诗羿、蓝浩宁、刘慎恒、张澳 |
| <07/01/21> | <1.1> | 完成测试结果及分析，缺陷清单和测试结论与建议 | 程可、邓诗羿、蓝浩宁、刘慎恒、张澳 |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 4

1.1 目的 4

1.2 范围 4

1.3 定义、首字母缩写词和缩略语 4

1.4 参考资料 4

1.5 概述 4

2. 测试概要 4

3. 测试环境 4

4. 测试结果及分析 4

4.1 需求覆盖率及缺陷分布 4

4.2 缺陷严重程度 5

5. 缺陷清单 5

5.1 功能性缺陷 5

5.2 非功能性缺陷 6

6. 测试结论与建议 6

系统测试报告

# 简介

## 目的

本文件的目的是根据系统测试用例的要求，对“小箱交云作业平台”系统进行系统测试，并记录相关情况说明。

## 范围

该测试报告覆盖项目包含的所有内容，包括网页前端页面，后端系统。

## 定义、首字母缩写词和缩略语

[1] Jmeter：Apache JMeter是Apache组织开发的基于Java的压力测试工具。用于对软件做压力测试，它最初被设计用于Web应用测试，但后来扩展到其他测试领域。 它可以用于测试静态和动态资源，例如静态文件、Java [小服务程序](https://baike.baidu.com/item/%E5%B0%8F%E6%9C%8D%E5%8A%A1%E7%A8%8B%E5%BA%8F/4148836)、CGI 脚本、Java 对象、数据库、[FTP](https://baike.baidu.com/item/FTP/13839" \t "_blank) 服务器， 等等。JMeter 可以用于对服务器、网络或对象模拟巨大的负载，来自不同压力类别下测试它们的强度和分析整体性能。另外，JMeter能够对应用程序做功能/[回归测试](https://baike.baidu.com/item/%E5%9B%9E%E5%BD%92%E6%B5%8B%E8%AF%95/1925732)，通过创建带有断言的脚本来验证你的程序返回了你期望的结果。为了最大限度的灵活性，JMeter允许[使用正则表达式](https://baike.baidu.com/item/%E4%BD%BF%E7%94%A8%E6%AD%A3%E5%88%99%E8%A1%A8%E8%BE%BE%E5%BC%8F/6555484" \t "_blank)创建断言。

## 参考资料

[1] 《小箱交云作业平台系统项目软件需求规约》

[2] 《小箱交云作业平台系统项目测试用例》

## 概述

测试报告包括测试的目的，范围，参考资料，测试概要，测试环境，测试的结果与分析，需求覆盖率及缺陷分布，缺陷清单，测试结论。其中测试结果与分析将从前后端两个方面来分别阐述。

# 测试概要

测试时间：2021年1月6日 – 2021年1月7日

测试地点：上海交通大学

测试人员：小箱交云作业平台开发小组

测试方法：详见《小箱交云作业平台系统测试用例》

测试内容：详见《小箱交云作业平台系统测试用例》

# 测试环境

软件环境及配置：Jmeter

硬件环境：笔记本电脑及租用服务器

笔记本电脑型号：联想拯救者Y7000

服务器：AmazonEC2系统， Unbuntu 18.04.4 LTS

内存：2 GB

# 测试结果及分析

[对于所做的各种测试，指出其结果。例如测试功能点数、测试用例数、缺陷数等。]

[给出主要测试结果分析的图形或表格表示，如需求覆盖率、缺陷分布图等。]

## 需求覆盖率及缺陷分布

*[需求覆盖率是指经过测试的需求/功能和软件需求规约中所有需求/功能的比值，通常情况下要达到100%的目标。需给出功能测试及非功能测试的需求覆盖情况。对于功能测试，需给出各个主要功能模块的需求覆盖率。如果需求覆盖率未达到100%可在备注项中说明原因及未测试内容。缺陷率是指本项缺陷占总缺陷数的百分比。请尽量使用图、表进行描述。]*

**表1 需求覆盖率及缺陷分布**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **测试内容** | | **测试**  **用例数** | **需求**  **覆盖率** | **缺陷数** | **缺陷率** | **备注** |
| **功能项** | **用户登录** | 4 | 100% |  |  |  |
| **用户注册** | 10 | 100% |  |  |  |
| **用户个人信息维护** | 2 | 100% |  |  |  |
| **教师用户增查改删课程及信息** | 4 | 100% |  |  |  |
| **教师用户为课程增删查改学生** | 4 | 100% |  |  |  |
| **教师用户布置作业** | 1 | 100% | 1 | 100% | 教师发布作业无法暂存 |
| **学生用户完成并且提交作业** | 1 | 100% | 1 | 100% | 学生的作业状态无法被刷新 |
| **教师用户批改作业** | 1 | 100% |  |  |  |
| **学生用户查看自己提交的作业被批改情况** | 1 | 100% |  |  |  |
| **用户之间私信交流** | 1 | 100% | 1 | 100% | 未实现 |
| **论坛讨论** | 1 | 100% | 1 | 100% | 未实现 |
| **管理员创建教师账号** | 1 | 100% |  |  |  |
| **管理员批量导入课程信息** | 1 | 100% |  |  |  |
| **照片清晰度检测** | 1 | 100% | 1 | 100% |  |
| **功能项小计** | 33 | 100% |  |  |  |
| **非功**  **能项** | **性能** | 3 | 100% |  |  |  |
| **易用性** | 3 | 100% |  |  |  |
| **可靠性** | 2 | 100% |  |  |  |
| **兼容性** | 4 | 100% |  |  |  |
| **界面测试** | 1 | 100% |  |  |  |
| **压力测试** | 1 | 100% |  |  |  |
| **部署测试** | 1 | 100% |  |  |  |
|  | **非功能项小计** | 15 | 100% |  |  |  |
|  | **总计** | 48 | 100% |  |  |  |

## 缺陷严重程度

*[描述缺陷按严重程度分布情况，包括不同严重程度缺陷个数及占缺陷总数百分比等。尽量使用图、表表示。]*

**表2 缺陷严重程度**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **严重程度** | **致命** | **严重** | **一般** | **细微** | **总数** |
| **缺陷个数** | 0 | 0 | 0 | 0 | 0 |
| **占缺陷百分比** | 0 | 0 | 0 | 0 | 0 |

# 缺陷清单

## 功能性缺陷

*[简要阐明功能缺陷汇总情况，并给出具体功能缺陷列表。]*

**表3 功能性缺陷列表**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序**  **号** | **缺陷**  **编号** | **严重程度** | **功能模块** | **缺陷标题** | **缺陷描述** | **测试用例编号** |
| 1 |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |

表中相关项说明：

1. 缺陷编号：为每个缺陷赋予一个唯一的编号，可以通过此编号对缺陷进行跟踪。例如：Bug001。
2. 缺陷严重程度：缺陷可以根据严重程度分为以下几种情况。
3. 致命（fatal）：致命的错误，测试执行直接导致系统死机、蓝屏、挂起、或是程序非法退出；系统的主要功能或需求没有实现。
4. 严重（critical）：严重错误，系统的次要功能点或需求点没有实现；数据丢失或损坏。执行软件主要功能的测试用例导致系统出错，程序无法正常继续执行；程序执行过于缓慢或是占用过大的系统资源。
5. 一般（major）：不太严重的错误，这样的缺陷虽然不影响系统的基本使用，但没有很好地实现功能，没有达到预期的效果。如次要功能丧失，界面错误，打印内容、格式错误，提示信息不太正确，或用户界面太差，简单的输入限制未放在前台进行控制，删除操作未给出提示，操作时间长等。
6. 细微（minor）：一些小问题，对功能几乎没有影响，产品及属性仍可使用。如软件的实际执行过程与需求有较小的差异；程序的提示信息描述容易使用户产生混淆；有个别错别字、 文字排列不整齐；辅助说明描述不清楚，显示格式不规范，长时间操作未给用户进度提示，提示窗口文字未采用行业术语，可输入区域和只读区域没有明显的区分标志，系统处理未优化等。
7. 功能模块：所测试并出现该缺陷的功能模块名称。
8. 缺陷标题：描述缺陷的标题。
9. 缺陷描述：对缺陷的详细描述。描述缺陷的重现步骤和问题所在，描述需清晰明了，使研发人员可以按照描述将缺陷重现出来。必要时可以放上相应的截图，以便于研发人员理解问题所在。

## 非功能性缺陷

*[简要阐明非功能缺陷汇总情况，并给出具体非功能缺陷列表。(根据所测系统的实际情况，可选择性地对系统进行必要的非功能性测试，包括系统的性能、可靠性、易用性、安全性、兼容性等。)请参考功能缺陷列表给出相关缺陷清单。]*

**表4 非功能性缺陷列表**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序**  **号** | **缺陷**  **编号** | **严重程度** | **测试类型** | **缺陷标题** | **缺陷描述** | **测试用例编号** |
| 1 |  |  | 性能 |  |  |  |
| 2 |  |  | 性能 |  |  |  |
| … |  |  | … |  |  |  |
| n |  |  | 可靠性 |  |  |  |

# 测试结论与建议

本次测试主要是针对于最后实现功能的功能测试。通过不断地测试与开发，修改了很多潜在的bug，对于项目的主逻辑部分测试覆盖率较高。测试为之后的开发提供了保障。