<风语>

测试报告

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 2019/07/10 | 1.0 |  | 袁健勇 |
|  |  |  |  |
|  |  |  |  |

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

测试报告

# 简介

## 目的

本报告用于记录与阐释对风语1.0的测试结果。

## 范围

相关项目：风语后端

## 定义、首字母缩写词和缩略语

Mc：管理员管理用户状态测试用例

Xf：查找函性能测试用例

Uu：用户更新个人信息测试用例

Us：用户发送验证码测试用例

Hp：函上传图片测试用例

## 参考资料

《Postman断言教程》

《Junit官方文档》

《Jmeter使用教程》

《软件工程导论PPt》

## 概述

其他文档：Admin, Hean, User三个文件夹和PostmanApiTest.html以及JmeterPressureTest.csv

其中，Admin, Hean, User三个文件夹分别是使用IntelliJ和Junit4的单元测试测试报告； PostmanApiTest.html是使用Postman进行接口测试生成的测试报告；

JmeterPressureTest.csv是使用Jmeter进行压力测试生成的测试报告

# 测试概要

测试时间：7.2-7.10

测试地点：软件学院

测试人员：项羽心、袁健勇

测试方法：单元测试、接口测试、压力测试

测试内容：各微服务代码逻辑的单元测试，微服务接口的集成测试，部署后的微服务群的压力测试

# 测试环境

测试硬件：4台公网阿里云服务器

测试软件：IntelliJ, Jmeter, Postman

测试环境：Java8, JUnit4

# 测试结果及分析

## 需求覆盖率及缺陷分布

**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** | **用户控制层** | 5 | 100% | 0 | 0 |  |
| **用户服务** | 4 | 100% | 0 | 0 |  |
| **Token服务** | 4 | 100% | 0 | 0 |  |
| **函控制层** | 6 | 100% | 0 | 0 |  |
| **函服务** | 9 | 100% | 0 | 0 |  |
| **管理员控制层** | 2 | 100% | 1 | 50% |  |
| **管理员服务** | 3 | 100% | 0 | 0 |  |
| **接口测试** | 24 | 100% | 0 | 0 |  |
|  | **功能项小计** | 57 | 100% | 1 | 1.7% |  |
| **非功**  **能项** | **性能** | 3 | 75% | 1 | 33.3% | 注册功能需要很多手机号和验证码，做不了并发测试 |
| **可靠性** | 3 | 100% | 0 | 0 |  |
| **安全性** | 10 | 100% | 3 | 30% |  |
|  | **非功能项小计** | 16 | 6% | 4 | 25% |  |
|  | **总计** | 63 | 100% | 5 | 7.9% |  |

## 缺陷严重程度

**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 | 5 | 5 |
| **占缺陷百分比** | 0 | 0 | 0 | 100% | 100% |

# 缺陷清单

## 功能性缺陷

**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 | Bug1 | 细微 | 管理员管理用户 | 管理员不能禁用管理员 | 超级管理员应该可以禁用低级管理员 | Mc3 |

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

**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 | Bug1 | 细微 | 性能 | 500并发查找函 | Jmeter对需要认证的查找函500个并发时，平均响应时间10.9s，错误率2.20% | Xf1 |
| 2 | Bug2 | 细微 | 安全 | 恶意用户修改他人信息 | 恶意攻击者可能会用自己的token和id通过postman等工具修改其他id用户的用户信息 | Uu3 |
| 3 | Bug3 | 细微 | 安全 | 多个验证码同时有效 | 用户注册时，如果多次发送验证码，那么在有效期内（5min），所有验证码都有效 | Us3 |
| 4 | Bug4 | 细微 | 节约 | 上传图片失败浪费空间 | 上传函时，如果其中某张图上传失败了，已上传的图片不会回滚，导致数据库的空间的浪费 | Hp3 |

# 测试结论与建议

* 性能上后端的架构有待改进，增强并发性和响应时间
* 安全上，漏洞还是比较多，可以从一个攻击者的角度去寻找漏洞和完善代码